

Федеральное государственное автономное образовательное учреждение  
высшего образования  
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»  
Институт управления бизнес-процессами и экономики  
Кафедра «Бизнес-Информатика»

УТВЕРЖДАЮ

Заведующий кафедрой

\_\_\_\_\_ А.Н. Пупков

« \_\_\_\_\_ » \_\_\_\_\_ 2018 г.

**БАКАЛАВРСКАЯ РАБОТА**

38.03.05.02. «Бизнес-информатика (Электронный бизнес)»

Разработка информационно-аналитического обеспечения для привлечения и  
удержания клиентов (на примере ООО «Велес Монтаж»)

Руководитель \_\_\_\_\_ к.т.н., доцент кафедры БИ. Р.И. Кузьмич

Выпускник \_\_\_\_\_ М.Е. Якимова

Нормоконтролер \_\_\_\_\_ Д.И. Ярещенко

Красноярск 2018

## РЕФЕРАТ

Выпускная квалификационная работа по теме «Разработка информационно-аналитического обеспечения для привлечения и удержания клиентов (на примере ООО «Велес Монтаж») состоит из введения, трех глав, заключения, списка использованных источников и приложений.

ИНТЕРНЕТ-МАГАЗИН, КЛИМАТИЧЕСКАЯ ТЕХНИКА, ПАНЕЛЬ АДМИНИСТРАТОРА, ИНФОРМАЦИОННО-АНАЛИТИЧЕСКОЕ ОБЕСПЕЧЕНИЕ.

Объектом исследования является ООО «Велес Монтаж».

Предмет исследования – информационно-аналитическое обеспечение привлечения и удержания клиентов.

Задачи работы:

- дать общую характеристику и обозначить тенденции развития ООО «Велес Монтаж»;
- выявить проблемы и способы их решения, возникающие при привлечении и удержании клиентов интернет-магазина;
- оценить эффективность существующего информационно-аналитического обеспечения интернет-магазина;
- провести выбор программных средств и разработать информационно-аналитическое обеспечение;
- оценить экономическую эффективность внедрения информационно-аналитического обеспечения.

Создание информационного обеспечения даст возможность предприятию повысить прибыльность и эффективность интернет-магазина, а также увеличить количество клиентов.

## СОДЕРЖАНИЕ

РЕФЕРАТ .....	2
ВВЕДЕНИЕ .....	5
1 Анализ основных проблем и механизмов их решения при привлечении и удержании клиентов ООО «ВЕЛЕС Монтаж» .....	7
1.1 Общая характеристика и тенденции развития ООО «ВЕЛЕС Монтаж» .....	7
1.2 Выявление проблем интернет-магазина, возникающих при привлечении и удержании клиентов .....	13
1.3 Обзор основных способов и технологий привлечения и удержания клиентов интернет-магазина .....	21
2. Обоснование разработки информационно-аналитического обеспечения ООО «ВЕЛЕС Монтаж» .....	29
2.1 Оценка эффективности существующего информационно-аналитического обеспечения интернет-магазина .....	29
2.2 Обоснование разработки информационно-аналитического обеспечения .....	36
2.3 Анализ вариантов реализации проекта .....	43
3. Разработка информационно-аналитического обеспечения для привлечения и удержания клиентов интернет-магазина ООО «ВЕЛЕС Монтаж» .....	52
3.1. Выбор и обоснование программных средств для информационно-аналитического обеспечения .....	52
3.2 Разработка информационно-аналитического обеспечения .....	61
3.3 Оценка экономической эффективности внедрения информационно-аналитического обеспечения .....	85
ЗАКЛЮЧЕНИЕ .....	91

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	92
ПРИЛОЖЕНИЕ А .....	96
ПРИЛОЖЕНИЕ Б.....	103

## ВВЕДЕНИЕ

Интернет стал неотъемлемой частью нашего быта, нашей работы и даже личной жизни. С каждым днем, проникновение интернет – технологий в нашу жизнь становится все более явным и глубоким.

За прошедшие годы, интернет стал наиболее эффективным средством рекламы и продвижения товаров и услуг, а также является одним из важнейших элементов современной цивилизации. Представить нашу жизнь без сети уже невозможно, она может удовлетворить все потребности современного человека: покупки, заключение деловых отношений, поиск клиентов, развлечения.

Интернет развивается стремительно и сегодня большинство компаний уже имеют свои сайты, так как пришло понимание, что иметь интернет – ресурс – это, скорее, необходимость.

Сайт – это виртуальный офис, работающий круглосуточно и без выходных. В любой момент пользователь может зайти и получить необходимую ему информацию.

Таким образом, целью выпускной квалификационной работы является привлечение и удержание клиентов интернет-магазина путем разработки информационно-аналитического обеспечения.

Для достижения поставленной цели необходимо выполнить следующие задачи:

- дать общую характеристику и обозначить тенденции развития ООО «Велес Монтаж»;
- выявить проблемы и способы их решения, возникающие при привлечении и удержании клиентов интернет-магазина;
- оценить эффективность существующего информационно-аналитического обеспечения интернет-магазина;
- провести выбор программных средств для реализации информационно-аналитического обеспечения;

- разработать информационно-аналитическое обеспечение;
- оценить экономическую эффективность внедрения информационно-аналитического обеспечения.

# **1 Анализ основных проблем и механизмов их решения при привлечении и удержании клиентов ООО «ВЕЛЕС Монтаж»**

## **1.1 Общая характеристика и тенденции развития ООО «ВЕЛЕС Монтаж»**

Полное наименование компании – Общество с ограниченной ответственностью «Велес Монтаж».

Сокращенное наименование компании – ООО «Велес Монтаж».

Директор компании – Пожарков Павел Васильевич.

Контактные данные (Телефон, E-mail):

- телефон 2-15-27-87;
- мобильный 8-923-366-41-02;
- e-mail veleskrs@mail.ru.

ИНН: 246518983062.

КПП: 246501001.

Адреса магазинов:

- Красноярский край, город Красноярск, улица Авиаторов, дом 27;
- Красноярский край, город Красноярск, улица проспект Молодежный, дом 25.

Виды деятельности по ОКВЭД [41]:

- 43.22 Производство санитарно-технических работ, монтаж отопительных систем и систем кондиционирования воздуха ;
- 43.21 Производство электромонтажных работ;
- 43.39 Производство прочих отделочных и завершающих работ;
- 43.31 Производство штукатурных работ;
- 33.12 Ремонт машин и оборудования;
- 33.14 Ремонт электрического оборудования;
- 33.20 Монтаж промышленных машин и оборудования;

- 42.21 Строительство инженерных коммуникаций для водоснабжения и водоотведения, газоснабжения;
- 42.22.1 Строительство междугородних линий электропередачи и связи;
- 43.13 Разведочное бурение;
- 43.29 Производство прочих строительно-монтажных работ;
- 43.32 Работы столярные и плотничные;
- 43.33 Работы по устройству покрытий полов и облицовке стен;
- 43.34.1 Производство малярных работ;
- 43.34.2 Производство стекольных работ;
- 43.91 Производство кровельных работ;
- 43.99.1 Работы гидроизоляционные;
- 43.99.3 Работы свайные и работы по строительству фундаментов;
- 43.99.4 Работы бетонные и железобетонные;
- 43.99.5 Работы по монтажу стальных строительных конструкций;
- 43.99.7 Работы по сборке и монтажу сборных конструкций;
- 43.99.9 Работы строительные специализированные, не включенные в другие группировки;
- 46.14.1 Деятельность агентов по оптовой торговле вычислительной техникой, телекоммуникационным оборудованием и прочим офисным оборудованием;
- 46.14.9 Деятельность агентов по оптовой торговле прочими видами машин и промышленным оборудованием;
- 46.15.3 Деятельность агентов по оптовой торговле электротоварами и бытовыми электроустановочными изделиями;
- 46.15.4 Деятельность агентов по оптовой торговле радио- и телеаппаратурой, техническими носителями информации;
- 46.43.1 Торговля оптовая электрической бытовой техникой;



- 46.51 Торговля оптовая компьютерами, периферийными устройствами к компьютерам и программным обеспечением;
- 46.71 Торговля оптовая твердым, жидким и газообразным топливом и подобными продуктами;
- 46.73.3 Торговля оптовая санитарно-техническим оборудованием;
- 46.73.6 Торговля оптовая прочими строительными материалами и изделиями;
- 46.74.2 Торговля оптовая водопроводным и отопительным оборудованием и санитарно-технической арматурой;
- 47.52.5 Торговля розничная санитарно-техническим оборудованием в специализированных магазинах.

Товары компании:

- системы автоматизации: Щиты управления, контроллеры, регуляторы температуры, частотные и трансформаторные регуляторы оборотов, приводы воздушных заслонок;
- вентиляторы осевые;
- крышные вентиляторы;
- прямоугольные вентиляторы;
- канальные вентиляторы смешанного типа;
- вентиляторы для промышленной и коммерческой вентиляции;
- фильтра и фильтрующие элементы;
- калориферы (нагреватели);
- оцинкованные монтажные элементы;
- решетки вентиляционные;
- потолочные диффузоры;
- решетки металлические вентиляционные;
- дверные решетки;
- потолочные решетки;
- тарельчатые клапаны;

- дверцы ревизионные;
- системы плоских и круглых ПВХ каналов;
- гибкие воздуховоды систем вентиляции и кондиционирования;
- шумоглушители.

Услуги компании:

- профессиональный подбор оборудования;
- монтаж бытовой и промышленной вентиляции;
- высококачественный монтаж любой климатической техники.

Компания «ВЕЛЕС» существует на рынке вентиляционного оборудования с 1998 года. Первый филиал был открыт 9 октября 2015 года.

Основным направлением компании является розничная и оптовая продажа вентиляционного оборудования отечественного и импортного производства.

Преимуществами компании являются:

- возможность изготовления любого вентиляционного оборудования под заказ в более короткие сроки;
- широкий ассортимент товаров российского и импортного производства;
- высокое качество товаров, подтвержденное лицензиями и сертификатами;
- предоставление услуг монтажа техники;
- предоставление услуг ремонта техники;
- ориентированность на промышленную и бытовую вентиляцию;
- регулярное расширение ассортимента товаров компании;
- оперативность работы компании;
- индивидуальный подход к каждому клиенту;
- наличие гибкой системы скидок.

Компания уже зарекомендовала себя как надежного и профессионального партнера и производителя. Пользуется популярностью у

постоянных клиентов, а за счет постоянного расширения предоставляемых услуг и прекрасного сервиса появляются все новые и новые клиенты.

Ориентированность на высокое качество продукции и удовлетворение любых желаний клиента являются ключевыми факторами успеха компании «Велес Монтаж».

Каждый месяц компания несет следующие расходы:

- хозяйственная деятельность (аренда помещения 15 тыс. руб., оплата интернета 1 тыс. руб., канцелярия 2 тыс. руб., телефонная связь 2 тыс. руб.);
- заработная плата работникам (3 сотрудника по 25 тыс. руб.);
- налоги на работников (2 500 на каждого сотрудника);
- расходы на рекламу (Яндекс-Директ 8 тыс. руб.).

На компанию достаточно сильно влияет курс валют – доллара и евро. Так как закупка оборудования и техники производится в условных единицах.

Однако, несмотря резкий скачек курса в последние годы компания не сдала своих позиций и сохранила уровень выручки. Что свидетельствует о том, что увеличился поток клиентов, появились новые заказы, несмотря на повышение цен. Это означает, что данной компании доверяют и тянутся к ней.

Помимо бытовых заказов, Организация ООО "Велес Монтаж" являлась поставщиком в 5 государственных контрактах. Заказчики представлены на рисунке 1.

В 2017 году открылся второй филиал компании на ул. Проспект Молодежный 25, что так же свидетельствует об успешной деятельности предприятия.

Проанализировав компанию ООО «Велес Монтаж», мы можем заметить, что она занимает свое значимое положение на рынке, несмотря на изменчивую ситуацию с курсом, сохраняются позитивные тенденции притока покупателей и развития организации.

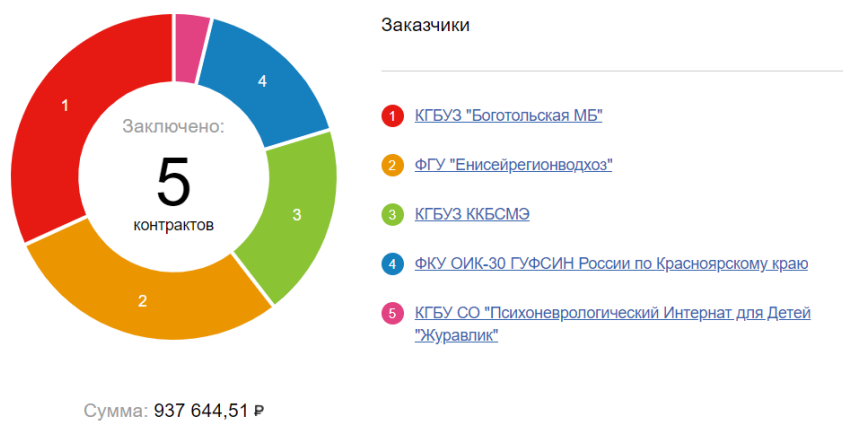


Рисунок 1 – Государственные заказчики ООО «Велес Монтаж»

Создание интернет – площадки и постоянное пополнение каталога продукции помогут привлечь новых клиентов, тем самым обеспечить более устойчивое положение на рынке, повысить прибыль и уровень доверия покупателей.

## **1.2 Выявление проблем интернет-магазина, возникающих при привлечении и удержании клиентов**

Итоговой целью любого интернет-магазина является клиент, который совершил покупку. Поэтому компании тратят время, деньги и усилия на привлечение в свой магазин новых покупателей и поддержание постоянных. Несколько лет назад, для того чтобы раскрутить интернет-магазин, нужно было просто подать рекламное объявление и ждать притока клиентов. А для удержания клиентов требовалось периодически сообщать об акциях или специальных предложениях. Сейчас же ситуация кардинально изменилась, но цель осталась та же. В условиях высокой конкуренции, компании должны постоянно повышать качество своего сервиса и предоставляемых товаров и услуг, а также создавать уникальные торговые предложения. Для того чтобы клиент остался, придется приложить немало усилий.

О том, насколько важно удержание клиента, писал в своей книге «Сервис как конкурентное преимущество» Джон Шоул (John Tschohl) [35]. Обратимся к статистике, приведенной автором:

- затраты на привлечение нового клиента обходятся в зависимости от отрасли от 5 до 10 раз дороже, чем на удержание уже существующего;
- сокращение оттока клиентов на 5 – 10 % может принести компании до 75% дополнительной прибыли;
- по мере насыщения рынка затраты на привлечение каждого нового клиента возрастают, а затраты на удержание сохраняются на стабильно невысоком уровне;
- большой процент постоянных клиентов обеспечивают высокую стабильность бизнеса и его привлекательность для внешних инвестиций.

Также наличие постоянных клиентов значительно сокращает затраты на маркетинг и рекламу.

Главный показатель – стоимость потерянного клиента. Стоимость потерянного клиента – это сумма денег, которую клиент мог бы потратить в

вашем интернет – магазине за максимально возможный период сотрудничества (в среднем, этот показатель ровняется 10 годам).

В борьбе за клиентов, интернет – магазин может столкнуться со многими проблемами. Некоторые из них мы сейчас рассмотрим.

Одной из проблем является устаревший дизайн сайта. Первое на что обращает внимание посетитель интернет ресурса – это на его внешний вид. Именно от дизайна сайта зависит останется он на нем, заинтересуется его содержимым или закроет вкладку. Нагромождение элементов и огромный список разделов, написанный мелким шрифтом, могут просто отпугнуть покупателя [11].

Неправильная SEO-оптимизация страниц. Каждая страница должна быть оптимизирована в соответствии с требованиями поисковиков: правильно прописаны метатеги, включены ключевые фразы в заголовки, подзаголовки и описание [38]. Основные мета-теги: Title, Description, Keywords. Первый содержит в себе заголовок, второй – описание страницы, а третий – основные ключевые слова, по которым ведется продвижение сайта. Таким образом мета-тег Keywords является семантическим ядром запросов пользователей. Он включает в себя определенный ряд слов, морфологических форм и словосочетаний, которые отражают смысл всего портала, и его разделов в частности. Ключевые слова подразделяются на высокочастотные, среднечастотные и низкочастотные. Высокочастотные отражают общий смысл текста, например «Купить дачу», среднечастотные являются уточнением, например «Купить дачу в Красноярске», а низкочастотные максимально уточняют и отражают запрос, например «Купить дачу в Красноярске недорого». Низкочастотные применяются на практике не часто, потому что запросы из списка низкочастотных слов появляются в системе довольно редко. Однако именно они приводят 100%-ную целевую аудиторию. При этом в семантическое ядро должны входить запросы с различной частотой. Многие поисковики отказались от мета-тега Keywords, например Google больше не учитывает данный тег. Это связано с

тем, что недобросовестные разработчики включали слишком много спама в данный тег, часто ключи не имели ничего общего с темой сайта. На данный момент мета-тег Keywords использует Яндекс, однако главное не перестараться с заполнением, иначе поисковые системы понизят вашу страницу или же весь сайт в поисковой выдаче. Ключевые запросы нужно правильно распределять между страницами, чтобы они стали релевантными. Возраст доменного имени для поисковых систем также является показателем надежности владельца сайта, так как продление домена из года в год требует вкладывания определенных средств. Однако при покупке старого доменного имени нужно быть очень осторожным. Вполне возможно, что это имя попадает под фильтры поисковых систем и вывести его в топ будет очень и очень сложно.

Также очень важно учитывать обновляемость сайта. Постоянное обновление содержимого сайта говорит о том, что он «живой» и содержит актуальную на сегодняшний день информацию. Помимо этого, следует принять во внимание тот факт, что новые страницы резко взлетают вверх в топе на некоторое время (примерно 1-4 недели), это связано с тем, что большинство запросов требуют новой свежей информации. Что является весомым аргументом в пользу регулярного обновления контента [6].

Так как в современном обществе сайты уже не являются просто площадками для торговли, нужно не только расширять каталог предоставляемых товаров и услуг, но и публиковать полезный материал, ответы на часто задаваемые вопросы. Так, покупатель мог просто искать информацию о товаре, а попав на ваш сайт, где расписана актуальная информация по данному товару и где можно приобрести этот товар, клиент явно будет нацелен на конкретную покупку.

В любом случае, пользователь должен попадать именно на ту страницу, которая максимально соответствует его запросу.

Недостаток информации о товаре при оформлении заказа. Бывает, что в процессе оформления заказа пользователь хочет уточнить условия доставки,

просмотреть информацию о товаре и убедиться еще раз, что выбрал именно то, что нужно. Но многие магазины дают такую информацию только в самом начале оформления покупки, а потом не показывают ее. Некоторые маркетологи считают, что в процессе оформления заказа не должно быть отвлекающих элементов. Однако информация о товаре, оплате и доставке очень важна для принятия решения о покупке. Также часто встретить оформление заказа в несколько страниц, что крайне неудобно, пользователю может элементарно наскучить переход по 3-4 страницам, чтобы совершить заказ. Чтобы этого избежать, следует сделать одностраничное оформление заказа, в котором вся информация о товаре и покупателе распределена по блокам и логически выстроена. Так же, во избежание каких-либо трудностей, следует использовать сервис обратной связи, чтобы можно было быстро и четко проконсультировать клиента по интересующему его вопросу.

Отсутствие кросс-продаж. Кросс-продажи или перекрестные продажи – это метод продаж, при котором покупателю предлагается купить также и другие продукты, дополняющие первый [6]. Это несущественная проблема, но внедрив кросс-продажи можно увеличить средний чек и повысить вовлеченность пользователей в процесс покупки. Так, например, при покупке фотоаппарата или телефона в дополнение можно предложить чехлы или зарядные устройства. Это обязательно заинтересует покупателя, а если на дополнительные товары будет предоставлена скидка, то можно не сомневаться в том, что покупка состоится. Так же перекрестные продажи снижают время поиска товаров клиента, потому что предлагают себя сами. Покупателю не придется обращаться к конкурентам или тратить больше своего времени, сайт сам поможет найти ему необходимые товары. Так же существенно снижаются издержки на поиск новых клиентов, так как реализация осуществляется уже существующим клиентам. Каждый продукт или услуга получают дополнительные точки продаж, что работает на увеличение оборотов. Перекрестные продажи нацелены на повышение уровня лояльности клиента и его преданности продавцу. В целом, высокая



эффективность кросс-продаж объясняется большими масштабами продаж и отсутствием дополнительных издержек [25].

Отсутствие отзывов. Отзывы имеют огромное значение для увеличения лояльности клиентов. Правильная работа с ними позволяет создать положительную репутацию, увеличить представленность бренда, повысить продажи. И напротив, ошибки в обработке отзывов приводят как к репутационным, так и финансовым потерям. Также следует соблюдать баланс положительных и отрицательных отзывов. Противоречащие друг другу отзывы отвлекают от совершения покупки. Если люди видят противоречивые комментарии, то они не будут разбираться, кто прав, а просто выберут другой товар. Тем не менее, это вовсе не означает, что нужно удалять негативные отзывы, ведь если на товар будут лишь положительные рекомендации, это может отрицательно сказаться на доверии клиентов. Помимо отзывов на сайте, существует множество проверенных сервисов для отзывов, таких как Flamp и Tulp. Там можно посмотреть не только сами отзывы от покупателей, но и обратную реакцию компаний и с какой скоростью они стараются разрешить все трудности с клиентами.

На данный момент существует тенденция покупать отзывы. Однако, покупатели это видят и понимают, у них сразу же пропадает доверие к компании, что влечет за собой потерю клиентов и ухудшение репутации компании, что для новичков в этом деле будет очень болезненно.

Отсутствие доступной контактной информации. Чтобы связаться с менеджерами или владельцами интернет - магазина, приходится перебрать огромное множество страниц, не каждый клиент согласен на такой долгий поиск. Размещать контактную информацию следует на каждой странице.

Отсутствие систем сбора информации и аналитики. Такие системы хранят информацию не только о покупках клиента и данных, которые он оставил, например, при регистрации, но и о том, какие страницы он просматривал, какие товары его заинтересовали. Благодаря этому можно информировать клиента или создавать для него специальные предложения

именно на интересующие его товары, что обязательно побудит желание совершить покупку. Яндекс.Метрика и Google Analytics – это самые популярные на сегодняшний день системы статистики и анализа посещаемости веб-сайтов. Функционал данных систем примерно одинаковый. Яндекс.Метрика – это интернет-сервис компании Яндекс, предназначенный для оценки посещаемости веб-сайтов, и анализа поведения пользователей. Позволяет собирать различную информацию о посещаемости сайта, о страницах, пользующихся наибольшей и наименьшей популярностью, также данный сервис группирует посетителей по определенным признакам, критериям и категориям, что может помочь в разработке особых предложений для разных групп посетителей. Данный сервис способен измерять конверсию сайта и рекламы, что помогает определить, как идет продвижение сайта и на что стоит обратить внимание. Очень важным фактором выбора Яндекс.Метрики является высокая оперативность предоставляемых сведений о посещаемости, отчеты обновляются каждые 5 минут. Google Analytics (сокращённо GA) – бесплатный сервис, предоставляемый Google для создания детальной статистики посетителей веб-сайтов. Является ограниченно бесплатной системой. Бесплатная версия действует, если посещаемость не достигает 5 млн уникальных просмотров за месяц. Данная система не только отслеживает количество уникальных посещений, но и отображает практически все действия, которые посетители выполняют на сайте. Отображает среднее время пребывания на сайте и наиболее и наименее популярные веб-страницы, как и Яндекс.Метрика. Отчеты могут включать в себя различные данные, в зависимости от индивидуальных настроек. Например, анализ ссылок и ключевых слов, по которым пользователи приходят на сайт и уходят с него, время пребывания на сайте, популярные страницы, группировка посетителей по различным критериям. Google Analytics сложнее Яндекс.Метрики для понимания и работы на начальных этапах. Так же обе системы дают разные показатели, в зависимости от того, в какую часть

помещен код, поэтому многие компании используют оба сервиса, для более углубленного анализа и продвижения.

Большие денежные затраты. На рекламу сайта, настройку Яндекс.Метрики или Google Analytics, обновление систем сайта, оптимизацию сайта, и так далее. Все это требует достаточно приличных затрат, которые новая компания может не потянуть. Также существуют рекламные площадки.

Например, 2Гис. Преимуществами данного продукта являются: удобный сервис, который также используется в качестве навигатора, работает без интернета, регулярно обновляется. Однако, чаще всего данную программу используют именно в качестве навигатора, но не как справочник для поиска контактов организации. Зависит от геолокации компании и сферы бизнеса, так как в некоторых случаях 2Гис может быть неоправданно дорогим. Официальные источники говорят нам, что из 2Гис звонят в 2,7 раз больше, чем в среднем из поисковых систем. При этом стоимость звонка в 2ГИС ниже в 3 раза. Есть три рекламных предложения с разной ценовой категорией: базовое (от 3 732 руб./мес.), включает в себя только рекламу, приоритет (от 7 656 руб./мес.), сюда так же добавляются логотип на карте и приоритетное размещение в рубрике, эксклюзив (от 20 193 руб./мес.), помимо всего вышеперечисленного сюда включается медийная реклама [50].

Рекламная сеть Яндекса (РСЯ) – это система размещения рекламы на сайтах, в мобильных приложениях, приложениях Smart TV и в видео. На этих платформах показываются рекламные блоки Яндекс-Директа. Яндекс не принимает к участию в Рекламной сети ресурсы с некачественным контентом или созданные специально для заработка на размещении рекламы. Исследования Яндекса, декабрь 2017 (аудитория поиска Яндекса и сайтов-участников РСЯ: среднее количество пользователей в день, которые видели больше 50% площади рекламного баннера в течение как минимум двух секунд) представлены на Рисунке 2.

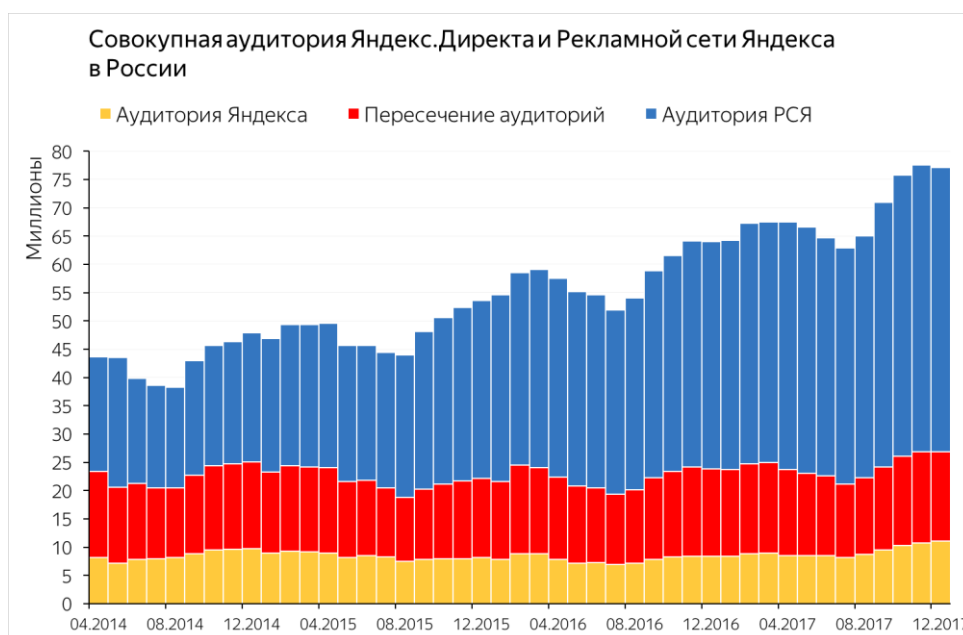


Рисунок 2 – Результаты исследования за 2017 год

В Яндекс-Директе цена рекламы абсолютно индивидуальна и рассчитывается в зависимости от ожиданий заказчика. На цену будут также влиять такие факторы как геолокация, спрос на товар, уровень конкуренции и другие. Если говорить грубо, то стоимость заказчик определяет сам. Начать кампанию можно при любом бюджете, начиная с 300 руб. Средний рекламный бюджет на месяц для большинства регионов России находится в пределах 30 тыс. руб.

Google AdWords также является достаточно популярным сервисом. Регистрация в Google AdWords бесплатна. Вы платите только тогда, когда пользователь нажимает на объявление и переходит на ваш веб-сайт или звонит в ваш офис. Другими словами – только когда реклама работает. Сервис также ориентирован на любую ценовую категорию и рассчитывается из нужд заказчика. Вы сами назначаете цену, которую вы готовы платить за 1 клик. В целом, если ежедневным бюджет компании составляет не менее 200 рублей, вы можете пользоваться данным сервисом и рассчитывать на бесплатную консультацию специалистов. Вы можете корректировать бюджет по мере необходимости и завершить кампанию в любой момент – неустойка при этом не взимается.

### **1.3 Обзор основных способов и технологий привлечения и удержания клиентов интернет-магазина**

Далее рассмотрим основные способы и технологии привлечения и удержания клиентов интернет-магазина.

Такой инструмент, как регистрация, является достаточно спорным. С одной стороны, каждый клиент хочет максимально быстро совершить покупку и обязательная регистрация может привести к тому, что клиент просто перейдет на другой сайт. Однако, данные, полученные при регистрации, могут упростить организацию повторных покупок, как для продавца, так и для покупателя. Пользователю не придется вновь вводить свои данные, он сможет просмотреть всю историю своих заказов, состояние заказов, составить лист желаний. А компания использует данные клиентов, чтобы создать базу клиентов, информировать их о каких-либо бонусных программах, узнать о степени удовлетворенности товаром и сервисом и в целом улучшить взаимодействие с клиентами. И тем и другим, регистрация дает свои плюсы и минусы. Идеальным решением данной ситуации будет регистрация по желанию. В конце, когда пользователь совершил покупку, можно предложить ему зарегистрироваться на сайте. В большинстве случаев, к этому времени пользователи успевают просмотреть большую часть сайта и заинтересоваться товарами, поэтому регистрируются. Существует так называемая скрытая регистрация. Она заключается в том, что для оформления заказа покупателю достаточно ввести свое имя и электронный адрес – после оплаты он уже будет являться зарегистрированным, активированным и авторизованным пользователем на сайте. Регистрационная информация сохраняется в куки браузера и отправляется почтой на указанный email. В последнее время набирает популярность еще один способ, самый удобный, регистрация через социальные сети. Для этого нужно будет просто нажать на иконку сети, через которую вы хотите

зарегистрироваться и все будет готово. При такой регистрации очень мала вероятность заспамления.

Персонализация сайта, которая использует данные, полученные при регистрации. Даже такая мелочь, как заполнение имени клиента в каком-либо письме или информировании и скидках, создает у покупателя ощущение, что это вовсе не рассылка, а персональное предложение.

E-mail маркетинг. Достаточно устаревший, но не менее действенный способ. Напоминает покупателю о необходимости вернуться на сайт компании, помогает информировать клиента о товарах, которые могут его заинтересовать. Здесь главное не перестараться, главное соблюдать границу между ненужной рекламой и полезными упоминаниями. В противном же случае пользователь просто заблокирует вашу компанию и письма либо перестанут приходить, либо автоматически будут отправляться в папку спам, что не принесет ничего хорошего ни магазину, ни клиенту. Полезной для клиента будет являться информация о действующих акциях и скидках, о персональных бонусах или же напоминание о забытой корзине. Также с помощью рассылки можно информировать покупателей об окончании гарантийного срока, если была приобретена какая-нибудь дорогая техника [5].

Использование социальных медиа, как и говорилось выше. Аккаунт в «Instagram», группа ВКонтакте, на сегодняшний день могут быть более действенными, чем любая реклама в интернете. Некоторые люди ведут свой бизнес исключительно в данных сетях. Такой выбор обосновывается тем, что «Instagram» и ВКонтакте наиболее популярны на данный момент социальные сети. Чаще всего для ведения такого бизнеса не требуется большего технического оснащения, кроме как телефон и мобильное приложение нужной сети. Недавно в ВКонтакте появилась функция перевода денежных средств, так что для совершения покупки не потребуется даже переходить в другое приложение. Социальные медиа являются хорошим способом повышения лояльности к компании. Они позволяют

взаимодействовать с клиентами ежедневно. При условии, что вы действительно будете вести свои аккаунты, так как длительное отсутствие обновлений приведет к тому, что клиенты попросту забудут о существовании компании. Если же ваша компания постоянно будет на виду, то вы сможете регулярно напоминать клиентам о себе. Так же социальные сети помогают выделить самых лояльных клиентов компании. Пользователи, которые поставили «like», и есть самые лояльные [34].

Организация скидок и акций. Способ, более ориентированный на привлечение клиентов, однако подойдет и для удержания. Часто можно встретить такую функцию, как скидка на первую покупку или же скидка при открытии магазина. В период скидок, клиенты активно раскупают товары, но, как только скидки заканчиваются, идет сильный отток клиентов. Поэтому этот способ подходит для привлечения. Здесь будет полезно наличие перекрестных продаж, так как это поможет извлечь больше прибыли от данных покупателей.

А вот создание эксклюзивных предложений ориентировано на удержание клиентов. Создание специальных условий после определенного количества покупок или суммы покупок, а также срока сотрудничества с компанией могут стимулировать клиентов продолжать совершать покупки. Здесь может пригодиться функция напоминания о заброшенной корзине или лист желаний клиента. Нужно будет ориентироваться именно на такие товары, у клиента также будет складываться положительное мнение о магазине и он будет чувствовать себя нужным.

Также пользуются огромной популярностью розыгрыши в социальных сетях. Конкурсы репостов привлекают внимание к бренду, увеличивают охват целевой аудитории и повышают лояльность. Многие люди не любят делать репосты, поэтому можно ограничиться лайком или комментарием. Конкурс с комментариями поможет сделать видимость большой клиентской базы и побудит интерес других пользователей.

Письма и звонки вежливости. Например, первая покупка – отличный повод связаться с состоявшимся клиентом, поблагодарить его за выбор, узнать о нем больше, а также получить обратную связь касательно степени его удовлетворенности. Следует регулярно обзванивать выборочных клиентов, тем самым показывая им их значимость для компании. Так же сбор информации с помощью данного способа может помочь устранить некоторые минусы компании, узнать конкретную причину неудовлетворенности клиента и совершить так называемую работу над ошибками. Обязательно следует фиксировать, с какими клиентами уже связывались сотрудники, навязчивость оставит после себя лишь негатив. А если такое будет регулярно повторяться, то пользователь может заблокировать компанию и вряд ли решит вновь воспользоваться ее услугами.

Организация обратной связи, посредством форума и написания отзывов. Покупатели должны видеть, что говорят о вашей компании. Положительные и отрицательные отзывы помогают повысить доверие к компании. Не стоит скрывать жалобы, на них нужно реагировать максимально быстро и решать возникшую проблему. Тем самым ваши клиенты будут понимать, что небезразличны вам. Многие не любят писать отзывы самостоятельно, они удовлетворены товаром и сроками доставки, но писать отзыв либо нет времени, либо нет желания. Здесь на помощь может прийти как раз звонок. Можно предложить клиенту запись отзыва с его слов. И для компании дополнительный отзыв всегда нужен и актуален и у самого клиента есть чувство, что он смог поделиться своим опытом с окружающими [5].

Очень важным инструментом является сервис компании. Компетентные консультанты, которые могут предоставить ответы на все интересующие вопросы. Быстрая реакция на жалобы. Отсутствие таких вещей может очень сильно расстроить клиента и вызвать у него нежелание повторно пользоваться вашими услугами. Регулярность в коммуникациях



является ключевым моментом в удержании клиентов. Нет регулярности – клиент будет забывать о компании и реагировать на предложения конкурентов. Если же компания будет редко писать и только высылать свои специальные предложения – клиент, скорее всего, будет воспринимать это как спам. Важно разбавлять свои рассылки об акциях, например элементарными поздравлениями клиентов с праздниками или же благодарностью за совершенный заказ.

Наличие контекстной рекламы. При грамотном подходе (необходимо принять решение, на каких ключевых словах, по которым осуществляется поиск, пользователю демонстрируется ваша реклама) контекстная реклама способна обеспечить максимально высокую конвертацию. Осуществляется посредством таких систем как Яндекс.Директ, Google AdWords и так далее.

SEO оптимизация. При поиске интересующего товара, клиенты редко выходят за пределы топ-10 поисковиков и переходят на вторую, третью и другие последующие страницы поиска. Как показывает статистика, более 95% человек останавливаются на первой. SEO оптимизация направлена как раз на то, чтобы вывести и удержать ваш сайт на наиболее высокой позиции в топе, а также на то, чтобы при запросе пользователь переходил на конкретную страницу товара [29].

Для организации SEO следует использовать системы аналитики, такие как Яндекс.Метрика или Google Analytics. Данные системы позволяют оценить целевой трафик, его величину, конкретные поисковые запросы, по которым осуществляются переходы и прочую полезную для SEO информацию.

Для поисковой оптимизации стоимость, оказывается в диапазоне от  $\approx 0,1$  рублей до  $\approx 20$  рублей за посетителя, что в несколько раз ниже аналогичной стоимости привлечения через системы контекстной рекламы. Этим объясняется привлекательность SEO как одного из основных каналов интернет-рекламы.

Автоматизации сбора данных и их аналитики. Для выстраивания надёжных взаимоотношений с покупателями потребуются точные и гибкие инструменты. Например, сервис Convead, который объединяет в себе аналитику и воздействие на посетителей интернет-магазина. Благодаря данному сервису, помимо сохранения информации о совершенных покупках, можно просмотреть всю историю действий клиента. Эти данные могут быть очень полезны для формирования целевого предложения каждому конкретному клиенту или таргетированной рассылки. Данный сервис также помогает осуществлять работу с заброшенными корзинами.

CRM система. Это не программный продукт, а модель бизнес-планирования, в центре которой находится клиент. Для реализации данного процесса собирается информация о каждом клиенте компании, а затем используется для построения результативных отношений с ним. Индивидуальная программа сотрудничества с каждым заказчиком помогает удержать существующих клиентов и расширить базу за счет привлечения новых [29]. CRM помогает автоматизировать процессы работы с заказчиками, вследствие чего компания имеет возможность предложить клиенту определенные продукты или услуги точно в то время, когда они ему необходимы. Такая концепция, когда на первый план ставится потребитель, а не товар, делает компанию конкурентоспособной на рынке.

Как правило, в список функциональных возможностей данного продукта входят такие опции:

- база контактной информации. В нее могут и должны быть включены все те, с кем сотрудники фирмы взаимодействуют постоянно. То есть клиенты, поставщики, подрядчики, партнеры и т. д. Эта информация позволяет экономить время и деньги. По сути, CRM – управление клиентами, которое осуществляется ненавязчиво, опираясь на множество факторов влияния;

- история сделок. Информация о том, что, когда и в каком количестве покупал клиент, в ряде случаев может помочь заключить ряд новых сделок.

Следовательно – повысить прибыль предприятия. Это – еще один аргумент в пользу CRM. Управление продажами с этим инструментом значительно упрощается;

- контроль над сделками. Имеется в виду, что клиент должен получать одинаково адекватную и актуальную информацию, независимо от того, присутствует ли на рабочем месте менеджер, который ведет его проект. Упорядоченные посредством CRM данные позволяют это делать просто и легко;

- база данных о мониторинге конкурентов. Чтобы бизнес был успешным, надо всегда знать, что и как делают конкуренты (акции, скидки, новинки), чтобы своевременно принять соответствующие меры;

- подготовка и рассылка коммерческих предложений (и других писем) в самой CRM. Все просто: существуют шаблоны, которые можно быстро скорректировать в соответствии с текущей потребностью, и, не теряя времени, отправить клиенту или партнеру. Это позволяет избавиться от необходимости тратить время на создание письма с нуля;

- календари, планировщики встреч и мероприятий. Этот функционал позволяет рационально спланировать время сотрудников, распределить часы использования переговорных комнат;

- формы отчетности. Наличие однообразных форм отчетности – отличный способ не только отслеживать текущее положение дел, но и составлять прогнозы;

- специальные функции, позволяющие сегментировать базы данных по клиентам и партнерам, предлагая каждому из них именно то, что нужно (скидки, семинары, новые товары, бонусы);

- автоматизированные рассылки, видео-встречи, вебинары и др. Эти функции нужны далеко не всем предприятиям, поэтому и покупать CRM с таким функционалом только потому, что это модно, не стоит;

Существует несколько видов CRM:

- операционные CRM. Главная задача операционных CRM-систем – повысить лояльность клиента в процессе непосредственного контакта с ним;
- аналитические CRM. Цель аналитических CRM – это анализ накопленной информации о клиентах и продажах для выстраивания более эффективной стратегии;
- коллаборационные CRM. Такие CRM налаживают коммуникации с клиентами для сбора обратной связи. Информация, полученная с их помощью, помогает скорректировать ассортимент товаров, ценовую политику, а также процесс обслуживания покупателей. Как таковых готовых CRM-систем этого типа нет: разрабатываются либо индивидуальные решения, либо используются существующие каналы связи, и информация из них фиксируется в основной CRM-программе;
- комбинированные CRM. В основном это операционные CRM с набором аналитических функций (отчеты по продажам, клиентам, эффективности менеджеров и т.д.), а также возможностью общения с клиентами (благодаря интеграции с сайтом, мессенджерами, социальными сетями и т.д.).

Цены на такие системы достаточно разные, но все выше среднего. Однако можно найти бесплатные версии с очень хорошим функционалом, которые предоставляются в пользование на время пробного периода, а некоторые на безвозмездной основе.

## 2. Обоснование разработки информационно-аналитического обеспечения ООО «ВЕЛЕС- Монтаж»

### 2.1 Оценка эффективности существующего информационно-аналитического обеспечения интернет-магазина

Для начала рассмотрим внешние особенности сайта и некоторый функционал.

Достаточно простой дизайн, не имеет каких-либо запоминающихся черт. Не обременен обилием анимации или слайдеров. Присутствует 1 слайдер на главной странице. В верхней строке можно увидеть контактные данные, кнопку обратной связи. Все это представлено на Рисунке 3.

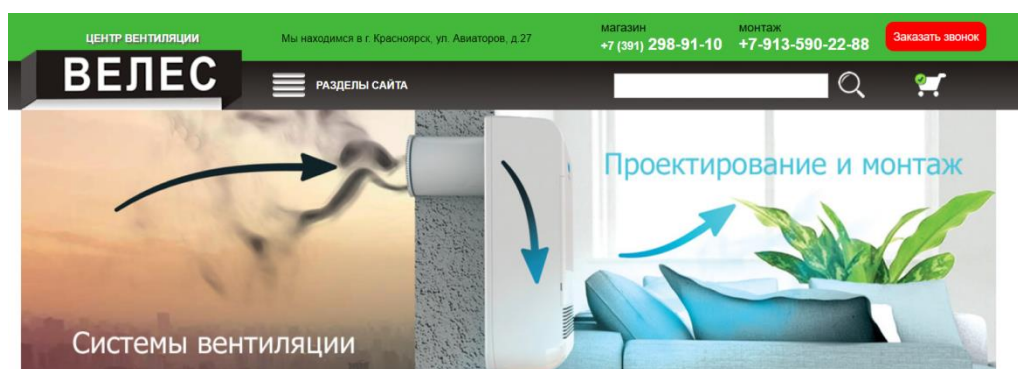


Рисунок 3 – Верхняя часть сайта старицы

Внизу страницы перечислены разделы сайта, по которым можно перейти, присутствуют ссылки на социальные сети, дополнены некоторые контактные данные. Меню с разделами сайта появляется лишь при нажатии на соответствующую кнопку. Здесь сразу же можно заметить следующий недостаток. При нажатии на кнопку «Разделы сайта», всплывает окно с разделами, которое полностью перекрывает контактную информацию в первой строке страницы, увидеть это можно на Рисунке 4.

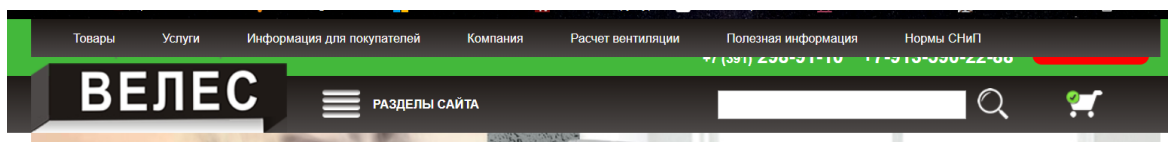


Рисунок 4 – Разделы сайта

Убрать данное меню нажав на ту же кнопку не получится, следует просто нажать на любое место на сайте и оно пропадет.

Ниже присутствует строка поиска. Поиск выполняется по введенным вами словам, однако, нет никаких фильтров или сортировки товара после поиска. Данная функция выдает все имеющиеся варианты, а покупатель не может настроить, например, ценовую категорию или интересующий его бренд. Использование поиска представлено на рисунке 5.

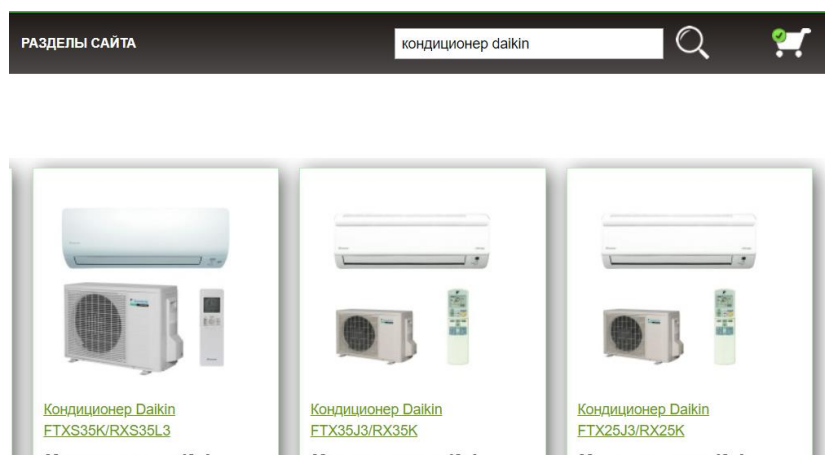


Рисунок 5 – Реализация поиска

При переходе на вкладку «Товары» в левой части появляется достаточно неудобное меню. Для того, чтобы покупателю выбрать, например, кондиционер, следует сначала пункт «Кондиционеры», далее выбрать вид кондиционера, далее выбрать марку и только потом клиент сможет увидеть интересующий его товар. Возможно, такая структура удобна, если клиент уже точно знает, что конкретно ему нужно. Но многих такое количество переходов будет отталкивать. Структура меню, на примере вкладки «Кондиционеры» представлена на рисунке 6.



Рисунок 6 – Структура меню на примере вкладки «Кондиционеры»

Когда клиент определился с заказом, он нажимает на кнопку заказать и ничего не происходит. Действие повторяется еще раз, и пользователь обновляет страницу, появляется цифра, обозначающая количество выбранных товаров. Но добавляются не все товары, возможно, некоторых просто нет в наличии, но сайт об этом никак не сообщает. Медленное реагирование на заказ может означать проблемы с хостингом или загруженностью кода. В итоге, клиент после двух нерабочих кликов попросту перейдет на другой сайт. При нажатии на значок корзины, появляется окно с выбранными товарами. Убрать данное окно можно только нажав на кнопку «Закрыть» и никак иначе, что так же не очень удобно.

Также, на сайте отсутствует обратная связь и система отзывов. Отсутствуют ссылки на ресурсы по типу Flamp, где можно увидеть, как компания взаимодействует с пользователями.

Далее перейдем к анализу сайта.

Оптимизация сайта. Оптимизация сайта – это улучшение его функционала и содержимого, а далее продвижение его в поисковых системах. Главная задача оптимизации – раскрутка сайта и привлечение

большого количества клиентов. На оптимизацию влияют как внутренние особенности сайта, такие как изображения, ключевые слова и так далее, так и внешние, например, количество ссылок на ресурс с уже популярных и раскрученных сайтов.

По данным PageSpeed Insights оптимизация сайта для ПК составляет 77% из 100. Это достаточно не высокий показатель, хотя и стремится к таковому, скорее средний [45].

Оптимизация сайта для мобильных телефонов низкая, всего лишь 57% из 100.

Видимость сайта.

Динамика видимости представлена на Рисунке 7.

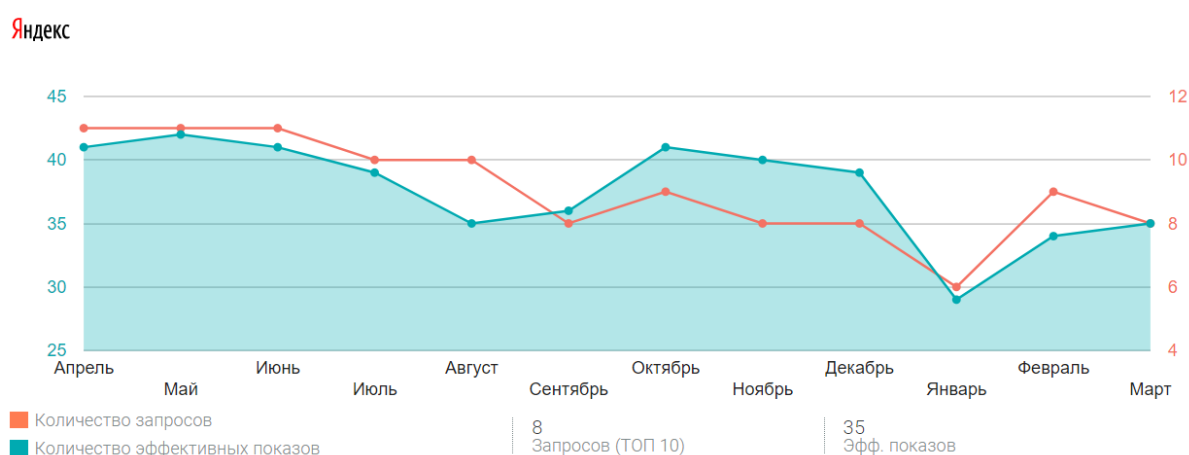


Рисунок 7 – Динамика видимости сайта компании ООО «Велес Монтаж»

Эффективные показы – это характеристика, учитывающая все три фактора оценки видимости: популярность поисковой системы, частота запроса и позиция в результатах поиска.

Количество эффективных показов в результатах поиска по запросу равно произведению текущей частоты запроса в данной поисковой системе на коэффициент видимости позиции.



Количество эффективных показов – это прогноз, сколько раз ссылка на анализируемый сайт попадет на глаза пользователям, задавшим поисковый запрос.

Рассмотрим динамику подробнее для поисковых систем Яндекс и Google. Данная информация представлена на Рисунках 8-9.

№	Запрос	Позиция	URL	Запросов в месяц	Эфф. показов
1	<a href="#">велес краснаярск</a>	1	<a href="#">/</a>	27	27
2	<a href="#">вентилятор улитка</a>	49	<a href="#">/catalogs/1/sections/20</a>	78	2
3	<a href="#">решетка 4са 600х600</a>	2	<a href="#">/catalogs/1/sections/2/prod...</a>	1	1
4	<a href="#">daikin краснаярск</a>	13 <sup>-1</sup>	<a href="#">/catalogs/1/sections/364</a>	1	1
5	<a href="#">купить кондиционер в краснаярске</a>	18	<a href="#">/catalogs/1/sections/99</a>	13	1
6	<a href="#">сантехническое оборудование это</a>	43	<a href="#">/pages/314</a>	17	1
7	<a href="#">светодиодные лампы в краснаярске</a>	30	<a href="#">/catalogs/1/sections/397</a>	21	1
8	<a href="#">расчет приточно вытяжной вентиляции</a>	46	<a href="#">/calculation-ventilation</a>	28	1
9	<a href="#">гофра для вентиляции</a>	49	<a href="#">/catalogs/1/sections/22</a>	19	1
10	<a href="#">жироуловитель сетчатый для вытяжки</a>	8	<a href="#">/catalogs/1/sections/115</a>	0	0

Рисунок 8 – Динамика запросов и показов поисковой системы Яндекс

№	Запрос	Позиция	URL	Запросов в месяц	Эфф. показов
1	<a href="#">велес краснаярск</a>	1	<a href="#">/</a>	27	27
2	<a href="#">жироуловитель для вытяжки</a>	4	<a href="#">/catalogs/1/sections/115</a>	8	6
3	<a href="#">крепеж для вентиляции</a>	14	<a href="#">/catalogs/1/sections/13</a>	25	3
4	<a href="#">магазин вентиляции</a>	12	<a href="#">/</a>	22	2
5	<a href="#">вентиляционные решетки промышленные</a>	3	<a href="#">/catalogs/1/sections/69</a>	1	1
6	<a href="#">жироуловитель для вытяжки цена</a>	6	<a href="#">/catalogs/1/sections/115</a>	1	1
7	<a href="#">фильтр жироуловитель для вытяжки</a>	8	<a href="#">/catalogs/1/sections/115</a>	1	1
8	<a href="#">частотный преобразователь для вентиляции</a>	10	<a href="#">/catalogs/1/sections/5</a>	1	1
9	<a href="#">вентиляция магазин</a>	20	<a href="#">/</a>	9	1
10	<a href="#">велес магазин</a>	25	<a href="#">/</a>	32	1

Рисунок 9 – Динамика запросов и показов поисковой системы Google

Проанализировав полученную информацию, можно сделать некоторые выводы. Большая часть эффективных показов достигается за счет запроса «Велес Красноярск», запроса с названием компании. Можно предположить, что по названию сайт компании ищут пользователи, уже знакомые с данной компанией, которые пользовались ее услугами. Это также обусловлено первой позицией в строке поиска.

Однако, при каких-либо конкретных запросах на товар, сайт находится на нижних позициях поиска, исходя из позиции, вероятность перехода на сайт очень мала, и переходы на него совершаются скорее из-за использования ссылок. Это может быть вызвано неправильно написанными ключевыми словами, плохой индексацией.

Динамика посещаемости.

График посещаемости представлен на Рисунке 10.



Рисунок 10 – График посещаемости сайта

Средняя посещаемость сайта – 1500 человек за месяц. Однако, в марте произошел резкий скачек, и количество заходов на данный сайт достигло отметки в 11 509. Однако продажи от этого не увеличились и компания ничего не получила. Следовательно, это были пустые заходы. Большие

средства компания вкладывает в продвижение на Яндексe, а в 2011, 2016 и 2017 на многих сайтах, стоящих в Яндексe на высоких позициях произошел настоящий бум заходов, количество заходов увеличивалось с 40 до 2000, однако, на сайт переходили вовсе не люди, а боты. Высока вероятность, что этот скачек также объясняется данным явлением. Это не несет в себе никакой угрозы, однако немного усложняет анализ.

Источники трафика сайта представлены на Рисунке 11.

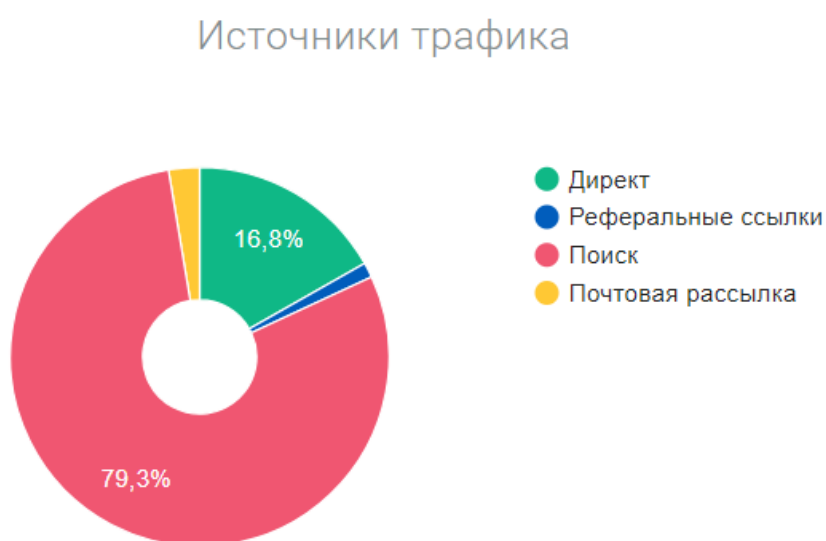


Рисунок 11 – Источники трафика

Подавляющее большинство переходов осуществляется за счет поисковых систем. Это хорошее значение. Но вместе с этим, слабой стороной является почтовая рассылка, а она является достаточно популярным способом продвижения, поэтому также стоит делать на нее акцент.

## **2.2 Обоснование разработки информационно-аналитического обеспечения**

Из анализа можно выявить следующие проблемные места, которые непременно следует устранить.

Структура товаров на сайте крайне неудобная, слишком большое количество переходов. А пользователь, который не знает, какой конкретный товар нужен, просто до него не доберется. Данную проблему можно изменить более удобной классификацией.



Необходимо создать расширенную функцию поиска, где можно сортировать товары по цене и маркам. Это ускорит нахождение товаров, а значит поможет сохранить клиентов, которые могли бы уйти из-за того, что не нашли нужные товар за 3 клика.

Отсутствие отзывов. Это очень большой минус, который необходимо устранить. Наличие отзывов формирует доверительные отношения, между продавцом и покупателем. И если отзывы на товар клиенты не пишут, возможно по причине того, что они не разбираются в технических характеристиках того же вентилятора, но свои функции он выполняет, то отзыв по работе монтажной или ремонтной бригады, они могут написать с большим удовольствием. Необходимо создать форум или систему для отзывов, и-за отсутствия такой связи компания может потерять достаточно большое количество клиентов. Так же, можно показать отзывы партнеров компании, чтобы показать ее надежность.

Далее, при оформлении заказа будет виден товар, который был добавлен в корзину, далее необходимо заполнить в поле имя, телефон, электронную почту, адрес доставки. Так же там будет предложено окно для добавления примечаний. Если покупатель заранее не ознакомился со статьей о способах оплаты и доставки на сайте, то при оформлении заказа такая информация предоставлена не будет, что крайне неудобно. Для перехода далее, клиенту достаточно заполнить лишь 2 графы «Имя» и «Телефон».

Далее появляется окно, в котором говорится, что консультант свяжется с вами для уточнения деталей. Реализация представлена на рисунках 12-14.

#### Оформление заказа

№	Наименование	Цена	Кол-во	Скидка	Сумма
1	 Сплит-система GALANZ AUS-09 H	14 334.40 руб	1 шт		14 334.40 руб
2	 Сплит-система GALANZ AUS-18 H	27 685.88 руб	1 шт		27 685.88 руб

Итого: 42 020 руб

Контактное лицо \*

Адрес доставки

Населенный пункт  Улица

Дом  Квартира / офис  Индекс  Ориентиры

Примечание

Оформить

Рисунок 12 – Окно оформления заказа

Контактное лицо \*

Адрес доставки

Населенный пункт  Улица

Дом  Квартира / офис  Индекс  Ориентиры

Примечание

Оформить

Рисунок 13 – Заполнение, необходимое для перехода на следующее  
ОКНО

## Заказ оформлен

Наши специалисты свяжутся с Вами в ближайшее время для его подтверждения.

Рисунок 14 – Окно оформленного заказа

Более расширенная форма оформления заказа, которая включает в себя выбор способа оплаты, выбора доставки, заказа монтажа может заметно упростить работу консультанта и обеспечить клиента большим количеством необходимой информации.

Еще одним минусом является отсутствие предложения о регистрации. Пользователь может зарегистрироваться, но только если сам найдет кнопку регистрации в самом конце страницы. Кнопка регистрации представлена на рисунке 15.

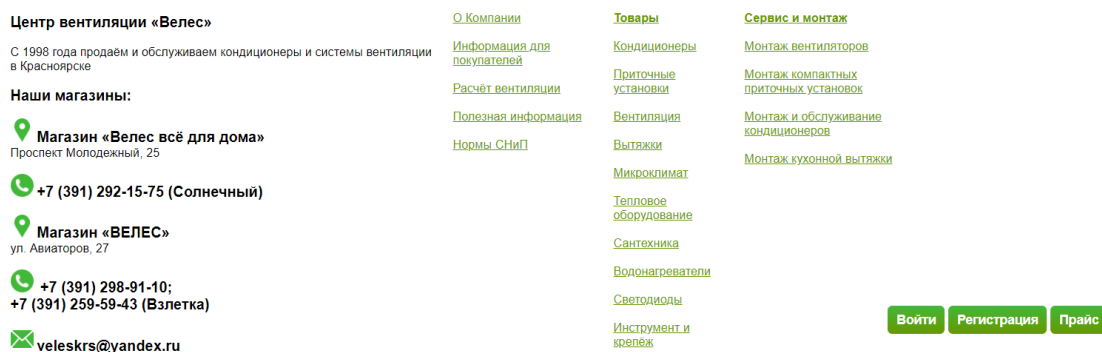


Рисунок 15 – Кнопка регистрации

После оформления заказа, следует предложить пользователю регистрацию. В большинстве случаев, клиенты регистрируются.

Следует улучшить организацию почтовых рассылок, так как количество переходов на сайт с их помощью очень мало. Помимо выгодных предложений, туда можно включать информирование об окончании

гарантийного срока обслуживания, так как имеем дело с достаточно затратной техникой.

Большая ставка сделана на поисковики Яндекс и 2Гис. Однако 2Гис чаще всего использую в качестве навигатора и он не может обеспечить переходы на сайт. Целесообразно вложить средства в раскрутку для Google поиска, так как из анализа можно заметить, что позиции сайта в данном поисковике достаточно низкие, а пользователи, как известно, редко выходят за пределы топ-3. Далее приведем примеры запросов, при которых компания не входит в данный топ и рискует остаться незамеченной. Результаты представлены в таблице 1.

Таблица 1 – результаты поиска Google.

Запрос	Позиция сайта
Велес	4
Кондиционеры Красноярск	Отсутствует на 2х первых страницах поиска
Очиститель воздуха Красноярск	Отсутствует на 2х первых страницах поиска
Монтаж вентиляции в Красноярске	Отсутствует на 2х первых страницах поиска
Проектирование вентиляции Красноярск	4
Кондиционер Daikin FTXB20C/RXB20C	10
Кухонные вытяжки купить Краноярск	8

Данный сайт использует технологию http. Однако в конце 2017 года Google обозначил свои реальные позиции относительно протоколов http и https:

- наличие HTTPS -протокола учитывается, как полноценный фактор ранжирования;

- по данным статистики <http://mozcast.com/features> на 24.10.2017 доля HTTPS-сайтов в ТОП-10 выдачи составила рекордные 69,5%, что можно увидеть на рисунке 16.

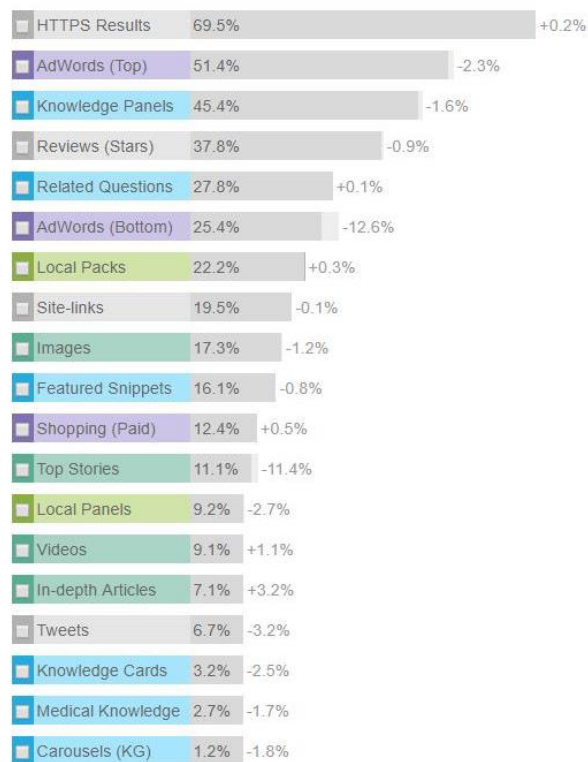


Рисунок 16 – Доля HTTPS сайтов

Представитель Google, Джон Мюллер заявил, что с большой долей вероятности HTTP/2 протокол ускорит сайт, если на нем уже установлен SSL/TLS сертификат.

Общеизвестный факт: время загрузки сайта является одним из важнейших факторов в ранжировании для обеих поисковых систем.

Браузер Google Chrome начал помечать HTTP-сайты, как небезопасные. Таким способом Google заботится о своих пользователях. Предупреждает их, что использовать данный сайт не рекомендуется, на нем могут «увести» персональные данные, пароли, данные банковских карт. Подтверждение представлено на рисунках 17-18.



Рисунок 17 – Предупреждение Google Chrome 1



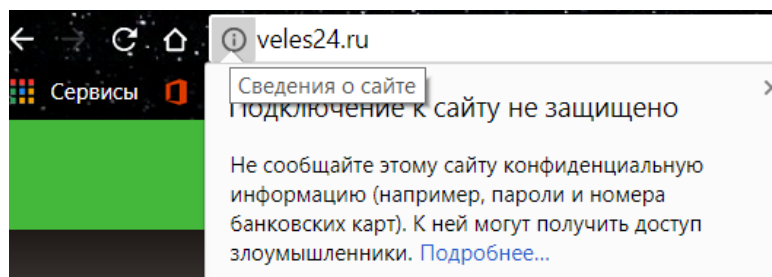


Рисунок 18 – Предупреждение Google Chrome 2

По данным LiveInternet [43], доля Яндекса в русском интернете упала до рекордно низких значений. Позиции Google же наоборот возрастают и уже почти сравнялись с Яндексом. Результат исследований представлен на рисунке 19.

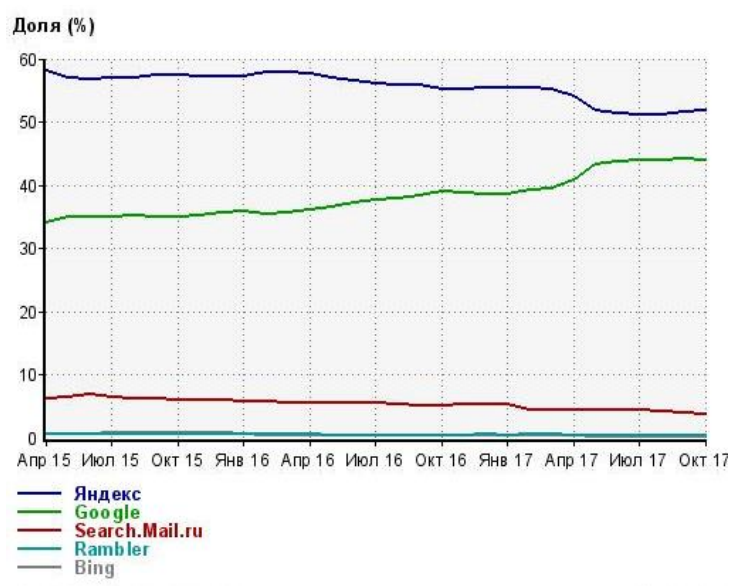


Рисунок 19 – Исследования LiveInternet

Тот факт, что большинство пользователей ищут через Яндекс, следует пересмотреть. Поэтому оптимизировать сайт компании «Велес монтаж» также следует под Google, это не отразится на его позициях в Яндексе.

Также низкие позиции в поиске могут быть обусловлены неправильным подбором ключевых слов. Следует улучшить и прописать ключевые слова различной частотности, так как это поможет повысить позиции компании в поисковиках по каким-либо узконаправленным запросам, а далее сохранять данную позицию.

Так же следует усовершенствовать оптимизацию сайта для ПК, а особенно для мобильных. В современном обществе люди больше всего взаимодействуют между собой и с коммерческими компаниями именно через телефон.

Для увеличения процента оптимизации можно совершить следующие действия:

- оптимизировать изображения (подобрать правильный формат и степень сжатия);
- удалить код JavaScript и CSS, блокирующий отображение верхней части страницы (Количество блокирующих скриптов на странице: 11. Количество блокирующих ресурсов CSS на странице: Замедляют отображение контента. Все содержание верхней части страницы отображается только после загрузки указанных далее ресурсов. Стоит отложить загрузку этих ресурсов, загружать их асинхронно или встроить их самые важные компоненты непосредственно в код HTML);
- сжать код JavaScript, что позволит сократить объем данных, чтобы ускорить загрузку, обработку и выполнение.

Исходя приведенных выводов можно заметить, что сайт подлежит усовершенствованию, причем достаточно глобальному. Имеется много проблемных мест, которые существенно влияют на работу сайта в целом, на количество клиентов и продаж. Для корректной работы сайта и увеличение показателей эффективности и продаж их следует устранить. Разбираться с исходным кодом, таким образом исправляя погрешности, не представляется возможным, так как данный сайт был заказан у отдельной компании, которая не предоставляет свои разработки в общее пользование. И в целом куда целесообразнее создать полностью новый и улучшенный сайт. Что и будет происходить далее.

## 2.3 Анализ вариантов реализации проекта

Сейчас сайт можно сделать 3 способами:

- использовать конструктор сайтов;
- использовать сервисы CMS;
- разработать сайт с нуля.

Конструктор сайтов – программно-реализованная сложная система для создания веб-страниц без знания языков программирования.

Сервис для создания сайтов работает из коробки – изначально идет в комплекте с хостингом, готовыми шаблонами и самим «конструктором» для управления и редактирования сайта. Рассмотрим наиболее популярные сервисы.

Конструктор сайта uKit. Российский конструктор сайтов от компании uKit Group. Позволяет без знания программирования собрать сайт-визитку с формой заказа, сайт-одностраничник или небольшой интернет-магазин, самостоятельно оптимизировать ключевые страницы для поиска и опубликовать сайт в сети интернет на собственном домене или домене третьего уровня.

Интерфейс современный, интуитивно понятный. Не имеет никакого кода и сложных настроек. Имеется множество готовых адаптивных шаблонов, представленных на рисунке 20, достаточно легко настраиваются. С ним и начинается создание сайта.

Для создания интернет-магазина в uKit следует подключить виджет «Товар магазина». С его помощью можно составить красивую витрину из любых товаров, задать опции (размер, вес, цвет), установить наценку. Корзина для добавления товаров на сайте появляется автоматически. В настройках виджета можно выбрать валюту, отредактировать форму заказа, задать параметры доставки и оплаты (оффлайн, PayPal, Яндекс.Касса и так далее). Пример страницы товаров представлен на рисунке 21.

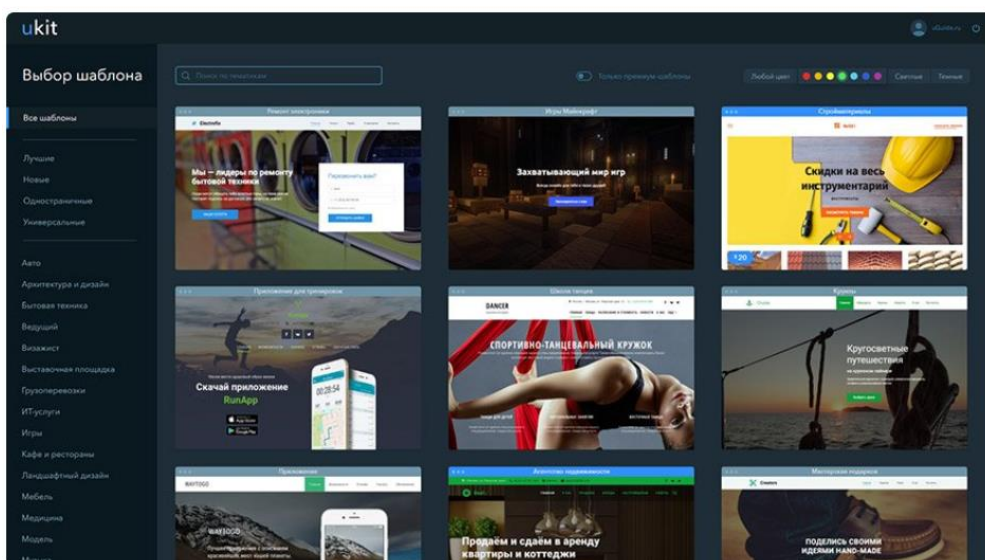


Рисунок 20 – Готовые шаблоны сайтов на uKit

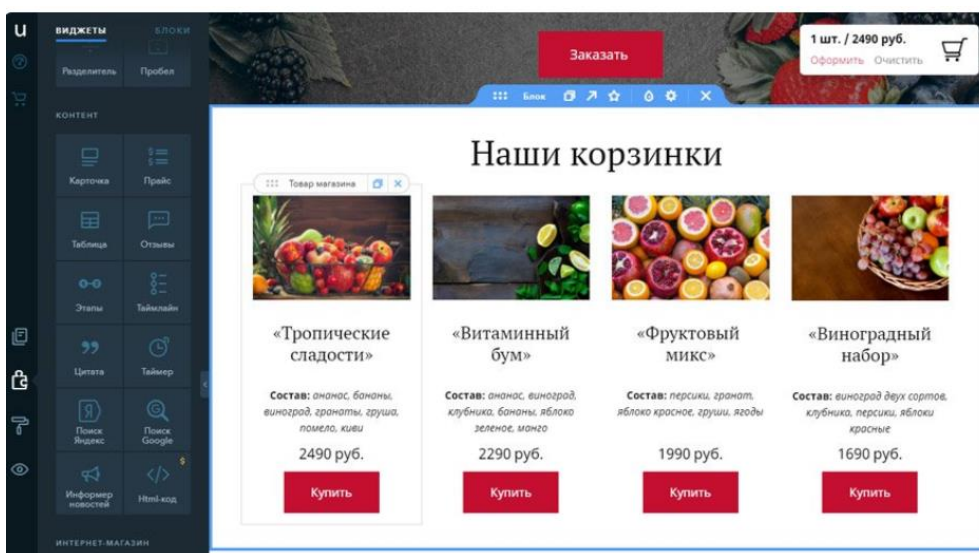


Рисунок 21 – Страница товаров на сервисе uKit

С помощью множества виджетов также можно настроить рассылки, шрифты и оформление текста, расположение элементов.

Предоставляется бесплатный, пробный период – 14 дней.

Далее имеется 4 тарифных плана:

- «Премиум» (\$4/мес или \$48/год) – все возможности конструктора, неограниченное количество страниц, места на хостинге и посетителей, прикрепление своих доменов (вида site.ru), без рекламы;

- «Премиум+» (\$8/мес или \$96/год) – дополнительно предоставляет расширенную статистику и доступ ко всем премиум-шаблонам;
- «Магазин» (\$9.6/мес или \$115/год) – открывает доступ к виджету Интернет-магазина, корзине и приёму онлайн-оплаты. Есть 14 дней бесплатного тестирования;
- «Про» (\$12/мес или \$144/год) – для профессионалов, вставка своего кода в сайт, возможность создания цветовых палитр вручную.

### Конструктор сайтов Wix.

Wix.com – международная облачная платформа для создания и развития интернет-проектов, которая позволяет конструировать сайты и их мобильные версии на HTML5 с помощью инструментов drag-and-drop. Является наиболее популярным во всем мире, но недешёвый на фоне своих аналогов.

Создание сайта также начинается с выбора шаблона. Примеры шаблонов представлены на рисунке 22. Шаблонов много, все разные, красивые и бесплатные. Все лежат в тематических категориях и имеют режим предпросмотра. Однако, сменить шаблон в процессе работы не получится: дело в большой разнице между шаблонами, которая не позволит одеть уже имеющийся контент в другой без нарушения структуры.

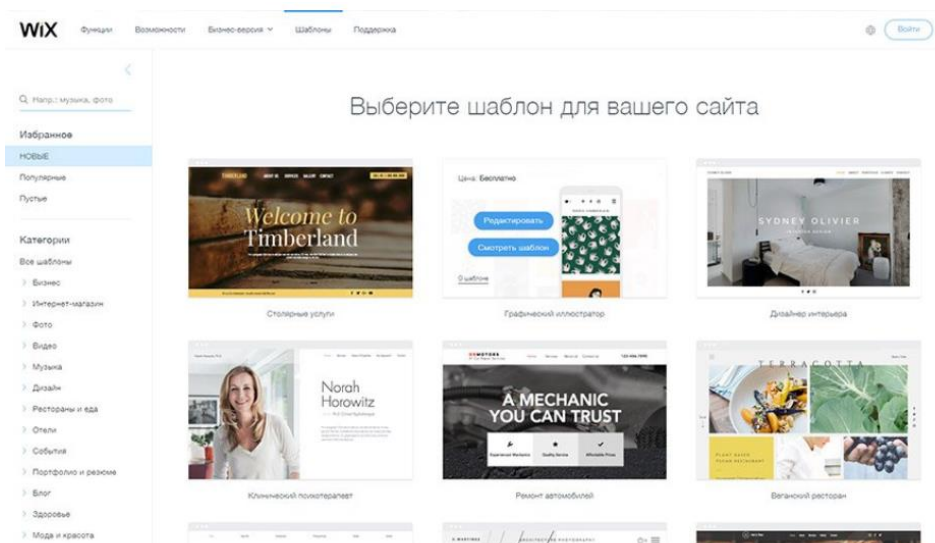


Рисунок 22 – Выбор шаблона сервиса Wix

Wix позволяет создавать сайты-визитки, лендинги, блоги, магазины.

В системе множество интересных виджетов, помимо стандартных, есть возможность вставить видео на фон, разные типы меню и прочее. Визуально впечатляющий на Wix сделать достаточно легко. Но такие перегруженные страницы будут очень медленно загружаться и какие-то особые элементы будут срабатывать не каждый раз. Пример возможного сайта представлен на рисунке 23.

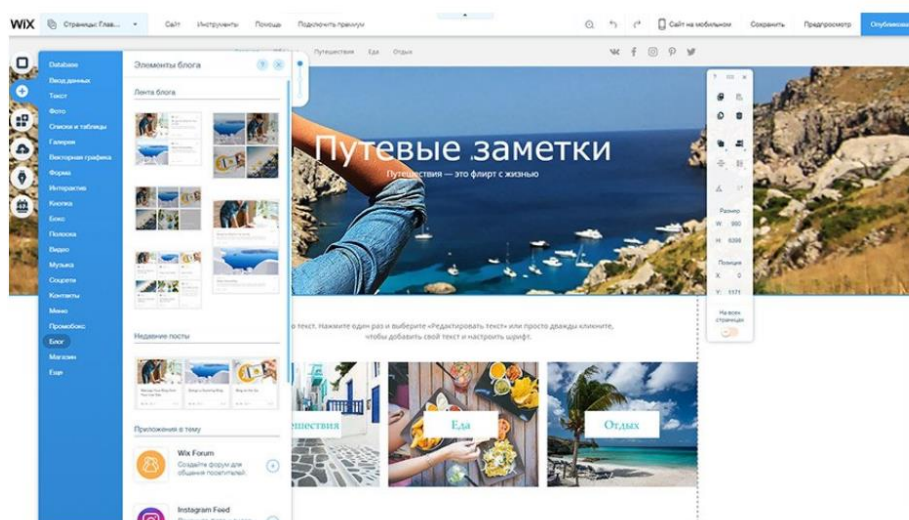


Рисунок 23 – Пример сайта, созданного на Wix

Wix предлагает следующие тарифные планы:

- «Connect Domain» (\$5.95/мес или \$49/год) – возможность подключить свой домен, получить премиальную техподдержку и 500 Мб дискового пространства, но не удаляет рекламу Wix;
- «Combo» (\$10.95/мес или \$99/год) – 3 Гб места, бесплатный домен, удаление рекламы, фавикон;
- «Unlimited» (\$15.95/мес или \$149/год) – увеличение дискового пространства до 10 Гб, конструктор форм, приложение Site Booster;
- «eCommerce» (\$19.90/мес или \$194/год) – 20 Гб места, возможность подключить корзину. Тариф для создания магазина;



– «VIP» (\$30/мес или \$294/год) – все преимущества eCommerce, плюс 10 E-mail рассылок/мес, а еще аудит сайта экспертами Wix (дизайн, юзабилити, seo).

CMS – система управления содержимым, информационная система или компьютерная программа, используемая для обеспечения и организации совместного процесса создания, редактирования и управления содержимым.

Рассмотри наиболее популярные сервисы.

WordPress – это самая популярная система управления контентом (CMS). По данным Web Technology Surveys, на этом «движке» по состоянию на 2015 год работает 23,8 % от общего числа существующих сайтов, а также 60,4 % сайтов, использующих CMS.

Создается и оплачивается доменное имя, далее создается база данных на сервере. На рисунке 24 представлен пример ввода регистрационных данных.

* Имя базы	<input type="text" value="a138803_baza_az"/>	Вы можете использовать маленькие латинские буквы a-z, цифры 0-9, символ подчеркивания _ и тире - Логин без префикса "a138803_" не должен быть длиннее 8 символов. Не используйте в этом поле свой логин в панель управления, он добавится автоматически.
* Логин	<input type="text" value="a138803_administ"/>	Вы можете использовать маленькие латинские буквы a-z, цифры 0-9, символ подчеркивания _ и тире - Логин без префикса "a138803_" не должен быть длиннее 8 символов. Не используйте в этом поле свой логин в панель управления, он добавится автоматически.
* Пароль	<input type="password" value="rd2Fc4ui2m"/> #	Пароль должен быть от 6 до 32 символов длиной.
Сложность:	<div style="width: 100%; height: 10px; background-color: green;"></div> Оптимальный	
<input type="button" value="Назад"/> <input type="button" value="Добавить"/>		
* - Обязательные поля		

Рисунок 24 – Регистрационные данные

Устанавливается WordPress, при регистрации необходимо указывать, будет ли будущий сайт индексироваться поисковыми системами. Пример установки показан на рисунке 25.

Требуется информация

Пожалуйста, укажите следующую информацию. Не переживайте, потом вы всегда сможете изменить эти настройки.

Название сайта

Имя пользователя   
Имя пользователя может содержать только латинские буквы, пробелы, подчёркивания, дефисы, точки и символ @.

Пароль, дважды  
Если вы оставите это поле пустым, пароль будет создан автоматически.  
  
  
  
Подсказка: Пароль должен состоять как минимум из семи символов. Чтобы сделать его надёжнее, используйте буквы верхнего и нижнего регистра, числа и символы наподобие ! " ? \$ % ^ & ;.

Ваш e-mail   
Внимательно проверьте адрес электронной почты, перед тем как продолжить.

Приватность ☒ Разрешить поисковым системам индексировать сайт.

Рисунок 25 – Установка WordPress

Важно сохранять пароли для входа и установки, так как их не так легко восстановить.

Далее выбирается шаблон или тема будущего сайта. Они представлены в самом WordPress, а также их можно скачать в интернете. Есть бесплатные шаблоны, однако за самые красивые и стильные придется заплатить.

Далее устанавливается тема и производятся общие настройки сайта. Настройки представлены на рисунке 26.

Сладкий клуб + Добавить Привет, admin Помощь

Общие настройки

Название сайта

Краткое описание   
Объясните в нескольких словах, о чём этот сайт.

Адрес WordPress (URL)

Адрес сайта (URL)   
Если вы хотите, чтобы адрес сайта [отличался от адресов](#) установки WordPress, введите адрес здесь.

Адрес e-mail   
Этот адрес используется в целях администрирования. Например, для уведомлений о новых пользователях.

Членство ☐ Любой может зарегистрироваться

Роль нового пользователя

Часовой пояс  UTC: 29.04.2015 10:36:05 Местное время: 29.04.2015 13:36:05  
Выберите город в том же часовом поясе, что и ваш.

Рисунок 26 – Общие настройки сайта



Затем подключаются различные плагины для создания слайдеров, особых шрифтов и всего, что необходимо для визуального наполнения. Также для создания интернет-магазина есть соответствующих плагины.

Если нужно расширить функционал сайта, разобраться в коде и программировать все же придется. Даже для того, чтобы русифицировать выбранный шаблон. Необходимо будет найти php-файл, найти нужную надпись и перепечатать ее русскими буквами.

Если требуется создать достаточно масштабный сайт, придется долго и упорно разбираться с CMS, а также подтянуть знания в веб-программировании, если не изучить их с нуля.

Не нужно объяснять, что функционал бесплатной версии достаточно мал и множество впечатляющих плагинов платные. Поэтому рассмотрим платные тарифные планы:

- «Личный». Стоимость 250 рублей в месяц. Пользовательское доменное имя, 6 Гб дискового пространства, отключена реклама;

- «Premium». Стоимость 550 рублей в месяц. Помимо всего функционала тарифа «Личный», сюда добавляются: премиум-темы без ограничений, дополнительные средства оформления, 13 Гб дискового пространства, возможность подключить стороннюю рекламу и получать от этого доход;

- «Business». Стоимость 1 499 рублей в месяц. неограниченный объём дискового пространства, SEO-инструменты, интеграция Google Analytics, возможность удалить сведения о бренде WordPress.com.

С помощью всех перечисленных сервисов конечно же можно создать интернет-магазин, но только небольшой и достаточно затратный.

Это уже не слишком подходит, для реализации поставленной задачи, но рассмотрим еще несколько доводов в пользу собственного написания сайта.

Почему нецелесообразно создавать сайты, с помощью конструкторов и различных сервисов:

- отсутствие полных прав на сайт. Созданный сайт является собственностью компании разработчика конструктора. Вы имеете доступ к аккаунту, к сайту, текстам и товарам;

- сложность переноса сайта. Если необходимо будет перенести сайт на свой хостинг, то это получится либо с большим трудом, либо вообще не получится;

- домен оформлен не на создателя. Как правило, у исполнителя будет домен третьего уровня. Иметь он будет вид: `site.domen.ru`. Но сейчас некоторые разработчики позволяют привязать свой домен к их системе за определенную плату;

- сложность полноценного продвижения. Из-за ограничения на большинство настроек, будет проблема с продвижением сайта в поисковиках. Только на платных тарифах, и то не везде есть возможность настроить файлы `.htaccess`, `robots.txt`, сделать редиректы и вручную прописать нужные мета-теги на всех страницах;

- ограниченный функционал. Владелец не будет иметь возможности сделать с сайтом все, что захочет;

- неуникальный дизайн сайта, так как приходится выбирать из готовых шаблонов;

- посторонняя реклама. Избавиться от нее можно только на платных тарифах и не на всех;

- медленная работоспособность и ограниченное место. Бесплатные и самые дешевые тарифы не предполагают мощных серверов и SSD дисков для хранения файлов. Так что по сравнению с обычным, сайт на конструктор будет медленнее.

Сайты на конструкторе идеально подходят в качестве первого сайта, а также для мелких и несерьезных проектов. Это отличное временное решение.

Но, если есть желание действительно отличаться от конкурентов, создать сайт лучше, чем у них, иметь полный доступ и высокие позиции в поисковиках, следует создавать сайт для себя на независимой платформе.

Исходя из всего вышесказанного, для реализации интернет-магазина будет выбран 3-й способ, а именно написание собственного сайта с использованием языков html, php, Mysql, фреймворка Notepad++ и сервисом ApacheMonitor.

### **3. Разработка информационно-аналитического обеспечения для привлечения и удержания клиентов интернет-магазина ООО «ВЕЛЕС-Монтаж»**

#### **3.1. Выбор и обоснование программных средств для информационно-аналитического обеспечения**

Перед тем как приступить к созданию сайта, требуется скачать и установить необходимые программные средства на свой персональный компьютер. Выбранные мной программные средства мы рассмотрим далее.

Для начала следует выбрать редактор кода. Данная программа необходима для всех разработчиков, т.к. позволяет работать с самим телом сайта, изменяя его внешний вид и функциональные возможности. Для создания веб-страницы следует написать ее код и сохранить в соответствующем формате, что без редактора сделать невозможно.

Самый простой редактор по умолчанию имеется на каждом компьютере – это блокнот. Однако, функционала данного продукта будет недостаточно для создания крупного проекта, поэтому следует обратиться к более специализированным программным средствам.

Для данной работы была выбрана последняя версия редактора Notepad++. Это один из самых популярных и наиболее простых в установке, использовании и использованию ресурсов компьютера редакторов.

Возможности программы, заявленные на официальном сайте:

- подсветка синтаксиса;
- сворачивание кода;
- автодополнение и автоматическое закрытие скобок и тэгов (если активировано)
- закладки;
- регулярные выражения для поиска и замены;
- запись и воспроизведение макросов;
- сравнение файлов;

- менеджер проектов;
- карта документа;
- переопределение любых горячих клавиш;
- резервное копирование сохраняемых файлов;
- трансформация текста при помощи подключённого плагина TextFX;
- поддержка и конвертирование кодировок ANSI, UTF-8 и UCS-2;
- блочное выделение текста, одновременное выделение нескольких

разных мест

- многострочное редактирование.

При установке дополнительных плагинов:

- шаблоны текста (сниппеты), вводимые с помощью сокращений (плагин SnippetPlus);
- FTP-менеджер (плагин NppFTP);
- Hex-редактор;
- автосохранение (при потере фокуса; через настраиваемый промежуток времени);
- проверка орфографии (с использованием GNU Aspell) ;
- симметричное и асимметричное шифрование текста (при установке плагина NppDarkCrypt);
- поддержка Zen Coding;
- поддержка автоматизации с помощью скриптов: Python, JScript, Lua, и других;
- поддержка сохранения в OneDrive и Dropbox.

Данный продукт позволяет подсвечивать различные языковые конструкции разными цветами, задавать разнообразные цветовые схемы для оформления рабочей области: фона и текста.

Также здесь доступна масса настроек, облегчающих разработку: автоматическое закрытие открытых скобок и кавычек, автоформатирование

текста, выделение ошибок и комментарии по поводу правильности написания конструкций.

Поддерживает открытие одновременно нескольких проектов.

Интерфейс редактора представлен на рисунке 27.

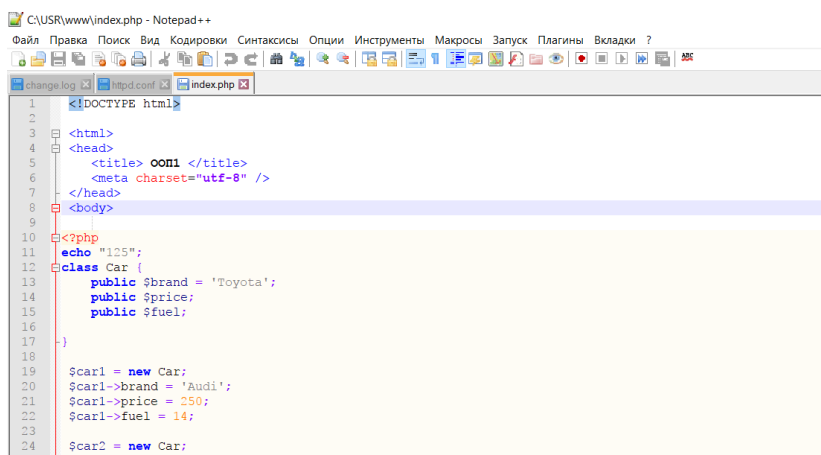


Рисунок 27 – Интерфейс редактора Notepad++

Далее следует выбрать локальный сервер.

Данная категория предназначена для запуска и отладки ресурса на локальном компьютере. Данная процедура обязательна при самостоятельной разработке веб-проекта, т.к. перед размещением в сети разработчику необходимо тщательно протестировать ресурс на всевозможные ошибки, работу его функционала и вид в различных браузерах пользователей.

Данный тип программ для разработки сайтов является «сборкой» других.

Для запуска сайта у себя на локальном ПК, установим следующие продукты:

- веб-сервер (Apache) – софт, принимающий запросы от клиентов, обрабатывающий их (в соответствии с кодом сайта) и выдающий ответ, который мы видим в окне Интернет-браузера;

– СУБД (MySQL) – совокупность инструментов для создания и управления созданной БД (базой данных) – хранилищем всей содержащейся на сайте информации;

– инструмент для работы с БД (PhpMyAdmin) – продукт, который устанавливается отдельно от СУБД и предоставляющий программный или веб-интерфейс для работы с БД;

– языковые средства (php) – некоторые языки веб-программирования, если они не идут в комплекте с каким-либо софтом, требуют установку своих библиотек, чтобы разработчики могли использовать их средства.

Веб-сервер Apache на данный момент самый популярный сервер, реализующий взаимодействие по протоколу HTTP, что является его основной функцией. С вероятностью 70% хостинг, на котором вы можете разместить свой сайт, будет использовать сервер Apache.

Главными преимуществами Apache являются стабильность, быстроедействие и гибкость, и обусловлены они модульной организацией.

Apache поддерживает огромное количество операционных систем и аппаратных платформ, и довольно часто его используют под управлением операционной системы Windows. Немаловажно и то, что данный web-сервер распространяется совершенно бесплатно.

После того, как вы скачали веб-сервер, вам необходимо произвести установку.

После корректной установки, на панели задач должна появиться иконка Apache, при нажатии на нее откроется контекстное меню, представленное на рисунке 28.

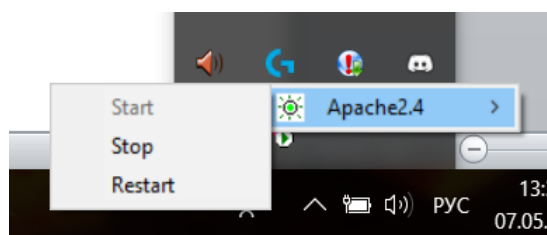


Рисунок 28 – Контекстное меню для управления сервером Apache

Для того чтобы проверить работает ли установленный сервер Apache, следует набрать в адресной строке браузера `http://localhost`. Если появилась страница с надписью «It works!» (представлена на рисунке 29), значит, установка Apache прошла успешно, и он функционирует правильно.



Рисунок 29 – Страница localhost, после корректной установки Apache

Далее выбирается СУБД.

MySQL является наиболее приспособленной для применения в среде web СУБД (системой управления базами данных). Для исполнения приложений клиента на большинстве хостинг-площадок провайдеры предоставляют небольшое количество ресурсов (как вычислительных, так и дисковых). Поэтому для данного применения необходима высокоэффективная СУБД, обладающая при этом высокой надежностью.

По всем этим причинам MySQL является стандартом в области СУБД для web, а теперь в ней развиваются возможности для использования ее в любых критичных бизнес-приложениях, то есть конкурирует на равных с такими СУБД таких производителей, как Oracle, IBM, Microsoft и Sybase.

Основные преимущества MySQL:

- многопоточность, поддержка нескольких одновременных запросов;
- оптимизация связей с присоединением многих данных за один проход;
- записи фиксированной и переменной длины;
- ODBC драйвер;



- гибкая система привилегий и паролей;
- гибкая поддержка форматов чисел, строк переменной длины и меток времени;

- интерфейс с языками C и Perl, PHP;
- быстрая работа, масштабируемость;
- совместимость с ANSI SQL;
- бесплатна в большинстве случаев;
- хорошая поддержка со стороны провайдеров услуг хостинга;
- быстрая поддержка транзакций через механизм InnoDB.

Далее устанавливается инструмент для работы с БД – PhpMyAdmin.

PhpMyAdmin – это приложение написанное на PHP и обеспечивающее полноценную, в том числе удаленную, работу с базами данных MySQL через браузер.

Так как phpMyAdmin позволяет во многих случаях обойтись без непосредственного ввода команд SQL, то работа с базами данных становится вполне посильной задачей даже для человека весьма поверхностно знакомого с MySQL.

Активное использование MySQL в веб-программировании обусловило его актуальность, а интуитивно понятный интерфейс в совокупности с широкой функциональностью и поддержкой более 60 языков (в т.ч. и русского) обеспечило ему популярность среди веб-разработчиков.

Преимущества PhpMyAdmin:

- возможность управлять СУБД MySQL без непосредственного ввода SQL команд;
- как панель управления RHPMyAdmin предоставляет возможность администрирования выделенных БД;
- интенсивное развитие;
- возможность интегрировать RHPMyAdmin в собственные разработки благодаря лицензии GNU General Public License и другие возможности.

Имеет понятный и простой интерфейс, главное окно программы представлено на рисунке 30.

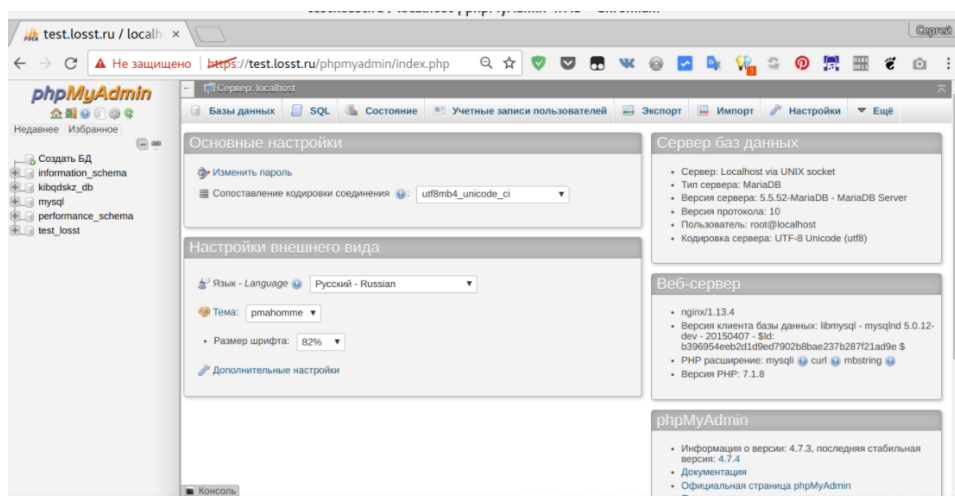


Рисунок 30 – Главное окно программы

Условно его можно разделить на несколько частей:

- боковая панель – находится слева, отображает вертикальный список доступных в данный момент баз данных, а также их таблиц. Можно развернуть каждую из них, чтобы посмотреть таблицы. Также есть быстрые кнопки – обновить, домой и так далее;

- главное меню – это полоска, которая находится под именем сервера. Именно с помощью происходит перемещение по различным возможностям программы, таким как базы данных, пользователи, экспорт и импорт и так далее;

- рабочая область – отображается рабочая информация, сейчас, в главном окне находится информация о сервере, в других же случаях содержимое базы данных или поле ввода текста.

Языковые средства.

Помимо HTML, необходимого для верстки сайта, будет использоваться язык PHP.

PHP – скриптовый серверный язык программирования общего назначения, интенсивно применяемый для разработки веб-приложений.

В настоящее время является одним из лидеров среди языков программирования, применяющихся для создания сайтов

Главным фактором языка РНР является практичность. РНР должен предоставить программисту средства для быстрого и эффективного решения поставленных задач.

Практический характер РНР обусловлен пятью важными характеристиками:

- традиционностью;

Язык РНР будет казаться знакомым программистам, работающим в разных областях. Многие конструкции языка позаимствованы из Си, Perl.

Код РНР очень похож на тот, который встречается в типичных программах на С или Pascal. Это заметно снижает начальные усилия при изучении РНР.

- простотой;

Сценарий РНР может состоять из 10 000 строк или из одной строки – все зависит от специфики задачи. Не нужно подгружать библиотеки, указывать специальные параметры компиляции. Механизм РНР просто начинает выполнять код после первой экранирующей последовательности (<?) и продолжает выполнение до того момента, когда он встретит парную экранирующую последовательность (?>). Если код имеет правильный синтаксис, он исполняется в точности так, как указал программист. РНР – язык, который может быть встроен непосредственно в html-код страниц, которые, в свою очередь будут корректно обрабатываться РНР-интерпретатором.

- эффективностью;

Очень важное преимущество РНР заключается в его «движке». «Движок» РНР не является ни компилятором, ни интерпретатором. Он является транслирующим интерпретатором. Такое устройство «движка» РНР позволяет обрабатывать сценарии с достаточно высокой скоростью.

- безопасностью;

В PHP реализованы механизмы безопасности, находящиеся под управлением администраторов; при правильной настройке PHP это обеспечивает максимальную свободу действий и безопасность. В стандартный набор функций PHP входит ряд надежных механизмов шифрования.

– гибкостью.

Поскольку PHP является встраиваемым (embedded) языком, он отличается исключительной гибкостью по отношению к потребностям разработчика. Хотя PHP обычно рекомендуется использовать в сочетании с HTML, он с таким же успехом интегрируется и в JavaScript, WML, XML и другими языками. Кроме того, хорошо структурированные приложения PHP легко расширяются по мере необходимости (впрочем, это относится ко всем основным языкам программирования). Нет проблем и с зависимостью от браузеров, поскольку перед отправкой клиенту сценарии PHP полностью компилируются на стороне сервера.

### 3.2 Разработка информационно-аналитического обеспечения

В процессе разработки информационно аналитического обеспечения необходимо выполнить следующие действия:

- разработать дизайн будущего сайта;
- сверстать сайт;
- создать движок для сайта;
- создать движок для admin-панели.

В процессе создания дизайна для сайта также необходимо разработать логотип для компании.

Цветовая гамма будущего сайта: преимущественно голубые и синие оттенки.

Развертка страницы: динамическая.

Типовая структура страницы представлена на рисунке 30.

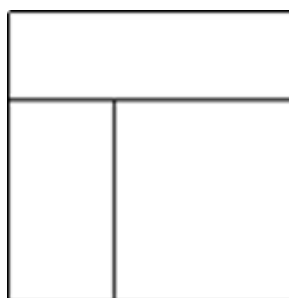


Рисунок 30 – Типовая структура страницы.

Самая часто используемая: Логотип, название и контакты левый верхний угол, под ним блок меню, корзина правый верхний угол, середина – основной контент.

Материалы, для наполнения сайта информацией, будут предоставлены заказчиком.

После разработки дизайна, приступаем к верстке сайта. В процессе используется блочная и табличная верстка. Листинг файлов верстки

представлен в Приложении Б. Листинг файлов стилей также представлен в приложении Б.

Далее приступаем к разработке движка для сайта. Это самая важная и объёмная часть работы.

Структура движка представлена на рисунке 31.



Рисунок 31 – Структура движка

Для начала, мы создаем базу данных в phpMyAdmin с разделами и товарами, которые будут представлены на сайте.

В данной базе данных есть 4 таблицы: таблица разделами, таблица с товарами, таблица со скидкой и таблица с заказами.

База представлена на рисунке 32.

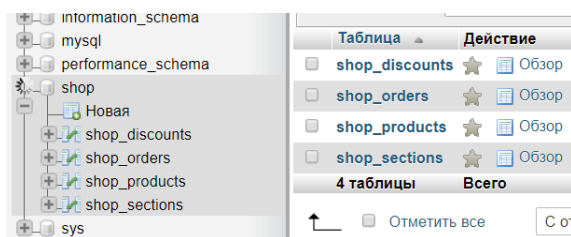
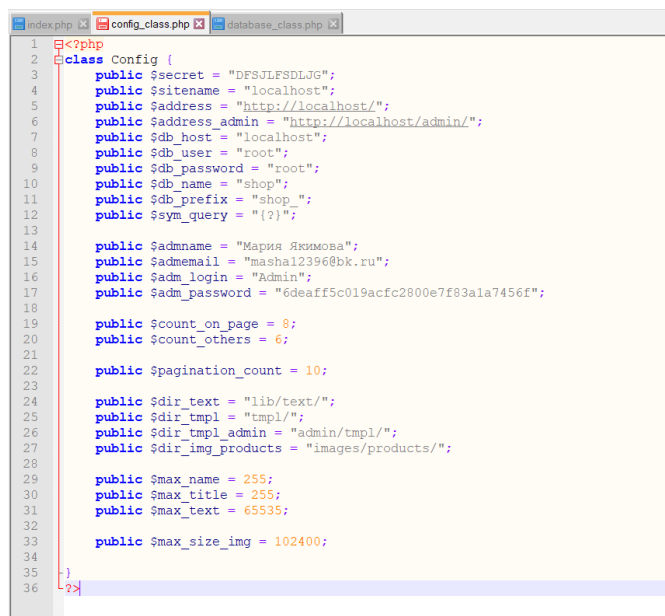


Рисунок 32 – База данных магазина «Велес-Монтаж»

Сначала создаем класс, файл с классом называется config\_class.php, для работы с настройками БД. Листинг файла config\_class.php представлен на рисунке 33.



```
1 <?php
2 class Config {
3     public $secret = "DFSJLFSDLJG";
4     public $sitename = "localhost";
5     public $address = "http://localhost/";
6     public $address_admin = "http://localhost/admin/";
7     public $db_host = "localhost";
8     public $db_user = "root";
9     public $db_password = "root";
10    public $db_name = "shop";
11    public $db_prefix = "shop_";
12    public $sym_query = "{?}";
13
14    public $admname = "Мария Якимова";
15    public $admemail = "masha12396@bk.ru";
16    public $adm_login = "Admin";
17    public $adm_password = "6deaaff5c019acfc2800e7f83a1a7456f";
18
19    public $count_on_page = 8;
20    public $count_others = 6;
21
22    public $pagination_count = 10;
23
24    public $dir_text = "lib/text/";
25    public $dir_tmpl = "tmpl/";
26    public $dir_tmpl_admin = "admin/tmpl/";
27    public $dir_img_products = "images/products/";
28
29    public $max_name = 255;
30    public $max_title = 255;
31    public $max_text = 65535;
32
33    public $max_size_img = 102400;
34
35 }
36
```

Рисунок 33 – Листинг файла config\_class.php

Далее создаем файл database\_class.php. В нем мы подключаем файл с конфигурацией (config\_class.php), далее создаем конструктор для вызова БД, в котором происходит инициализация БД, при этом используется лишь 1 объект. Так же в этом файле прописываются методы, которые позволят извлечь несколько строк, таблицы целиком, конкретную строку таблицы или конкретное значение ячейки. Фрагмент кода представлен на рисунке 34, полный листинг файла можно увидеть в приложении Б.

Далее создаем класс format\_class.php, где у нас будут форматироваться данные, а именно дата и время, обеспечиваться рекурсивная обработка запросов. Фрагмент файла представлен на рисунке 35, с полным листингом можно ознакомиться в приложении Б.

Далее создаем класс url\_class.php. Данный класс будет заниматься хранением адресов наших страниц. Фрагмент кода представлен на рисунке 36, полный листинг можно посмотреть в приложении Б.

```

1 <?php
2 require_once "config_class.php";
3
4 class DataBase {
5
6     private static $db = null;
7     private $config;
8     private $mysqli;
9
10    public static function getDB() {
11        if (self::$db == null) self::$db = new DataBase();
12        return self::$db;
13    }
14
15    private function __construct() {
16        $this->config = new Config();
17        $this->mysqli = new mysqli($this->config->db_host, $this->config->db_user, $this->config->db_password, $this->config->db_name);
18        $this->mysqli->query("SET NAMES 'utf8'");
19    }
20
21    private function getQuery($query, $params) {
22        if ($params) {
23            for ($i = 0; $i < count($params); $i++) {
24                $pos = strpos($query, $this->config->sym_query);
25                $arg = "i,$this->mysqli->real_escape_string($params[$i])."";
26                $query = substr_replace($query, $arg, $pos, strlen($this->config->sym_query));
27            }
28        }
29    }
30

```

Рисунок 34 – Фрагмент кода database\_class.php

```

1 <?php
2 require_once "config_class.php";
3
4 class Format {
5
6     private $config;
7
8     public function __construct() {
9         $this->config = new Config();
10    }
11
12    public function ts() {
13        return time();
14    }
15
16    public function date($t) {
17        return date("Y-m-d H:i:s", $t);
18    }
19
20    public function time($ts) {
21        return date("d.m.Y", $ts);
22    }
23

```

Рисунок 35 – Фрагмент класса format\_class.php

```

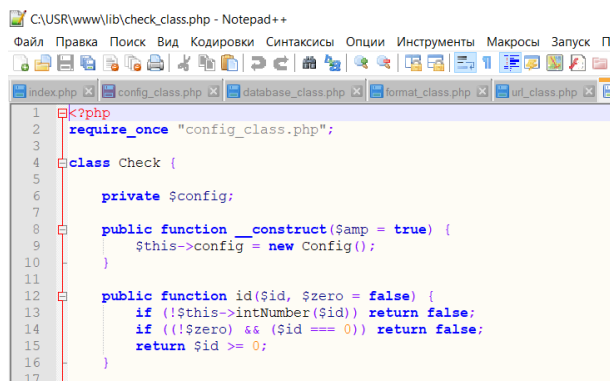
1 <?php
2 require_once "config_class.php";
3
4 class URL {
5
6     protected $config;
7     protected $amp;
8
9     public function __construct($amp = true) {
10        $this->config = new Config();
11        $this->amp = $amp;
12    }
13
14    public function getView() {
15        $view = $_SERVER["REQUEST_URI"];
16        $view = substr($view, 1);
17        if (($pos = strpos($view, "?")) !== false) {
18            $view = substr($view, 0, $pos);
19        }
20        return $view;
21    }
22

```

Рисунок 36 – Фрагмент кода url\_class.php



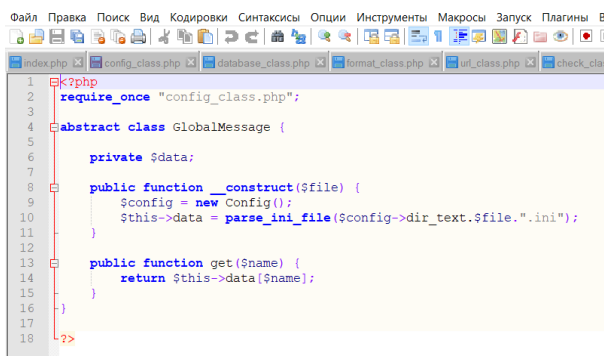
Далее создаем класс `check_class.php`. Данный класс занимается проверкой входных данных. Фрагмент кода представлен на рисунке 37. Полный код находится в приложении Б.



```
1 <?php
2 require_once "config_class.php";
3
4 class Check {
5
6     private $config;
7
8     public function __construct($amp = true) {
9         $this->config = new Config();
10    }
11
12    public function id($id, $zero = false) {
13        if (!$this->intNumber($id)) return false;
14        if (!$zero && ($id == 0)) return false;
15        return $id >= 0;
16    }
17 }
```

Рисунок 37 – Фрагмент файла `check_class.php`

Далее мы создаем класс `globalmessage_class.php`. Данный класс работает с различными системными сообщения, служебными письмами, email-письмами и т.д. Листинг файла представлен на рисунке 38.



```
1 <?php
2 require_once "config_class.php";
3
4 abstract class GlobalMessage {
5
6     private $data;
7
8     public function __construct($file) {
9         $config = new Config();
10        $this->data = parse_ini_file($config->dir_text.$file.".ini");
11    }
12
13    public function get($name) {
14        return $this->data[$name];
15    }
16 }
17
18 >
```

Рисунок 38 – Листинг файла `globalmessage_class.php`

Вместе с этим классом, создаем файлы `messages.ini`, где содержатся сообщения об ошибках, `page_messages.ini`, который будет содержать полноценные сообщения на страницу, и `emails.ini`, который будет содержать сообщение на email клиента. Листинг данных файлов будет представлен на рисунках 39 – 41. В данных файлах содержатся html-коды.

```

C:\USR\www\lib\text\messages.ini - Notepad++
Файл Правка Поиск Вид Кодировки Синтаксисы Опции Инструменты Макросы Запуск Плагины
index.php config_class.php database_class.php format_class.php url_class.php ch
1 UNKNOWN_ERROR=Неизвестная ошибка
2 ERROR_NAME=Некорректное ФИО
3 ERROR_PHONE=Некорректный телефон
4 ERROR_EMAIL=Некорректный e-mail
5 ERROR_ADDRESS=Некорректный адрес
6 ERROR_PRICE=Некорректная цена
7 ERROR_NOTICE=Некорректное примечание
8 ERROR_DELIVERY=Вы не указали способ доставки
9 ERROR_AUTH=Неверный логин и/или пароль
10 SUCCESS_ADD_PRODUCT=Новый товар добавлен!"
11 SUCCESS_EDIT_PRODUCT=Товар успешно изменён!"
12 SUCCESS_DELETE_PRODUCT=Товар успешно удалён!"
13 ERROR_IMG=Некорректное имя изображения
14 ERROR_TYPE_IMG=Некорректный тип изображения
15 ERROR_SIZE_IMG=Превышен максимальный размер изображения
16 ERROR_EXISTS_IMG=Изображение с таким именем уже существует
17 ERROR_TITLE=Некорректное название

```

Рисунок 39 – Фрагмент файла messages.ini

```

C:\USR\www\lib\text\page_messages.ini - Notepad++
Файл Правка Поиск Вид Кодировки Синтаксисы Опции Инструменты Макросы Запуск Плагины Вкладки ?
index.php config_class.php database_class.php format_class.php url_class.php check_class.php globalmessage_class.php messages.ini page_
1 SUCCESS_PAY_TITLE=Оплата принята!"
2 SUCCESS_PAY_TEXT=Оплата успешно принята. Дождитесь звонка менеджера.
3 FAIL_PAY_TITLE=Ошибка при оплате!"
4 FAIL_PAY_TEXT=Оплата не прошла, попробуйте повторить процесс. Если деньги уже сняли, то обратитесь к администрации.

```

Рисунок 40 – Файл page\_messages.ini

```

index.php config_class.php database_class.php format_class.php url_class.php check_class.php globalmessage
1 ORDER_TITLE=Ваш заказ
2 ORDER_TEXT=Здравствуйте!"<br /><br />Ваш заказ принят!"<br />
3 <br />Данные:<br />Ваш заказ: %products%<br />ФИО: %name%<br />Телефон: %phone%<br />
4 E-mail: %email%<br />Адрес: %address%<br />
5 Примечание: %notice%<br />
6 <br />Стоимость заказа: %price% рублей<br />
7 В ближайшее время с Вами свяжутся наши менеджеры для подтверждения заказа.<br />
8 <br /><i>С Уважением, %sitename%.</i>

```

Рисунок 41 – Файл emails.ini

Для работы с файлом mesaages.ini создаем класс message\_class.php. Код представлен на рисунке 42.

```

index.php config_class.php database_class.php format
1 <?php
2 require_once "globalmessage_class.php";
3
4 class Message extends GlobalMessage {
5
6     public function __construct() {
7         parent::__construct("messages");
8     }
9 }
10

```

Рисунок 42 – Код файла messages.ini

Так как файл messages.ini не имеет никаких префиксов и заголовков, а два других файлы имеют, нас следует создать еще 2 класса с работы с этими файлами. Далее создаем класс complexmessages\_class.php. Является

расширением класса `globalmessages_class.php`. Содержит два метода, которые возвращают заголовок и сам текст. Код `complexmessages_class` представлен на рисунке 43.

```
1 <?php
2 require_once "globalmessage_class.php";
3
4 abstract class ComplexMessage extends GlobalMessage {
5
6     public function getTitle($name) {
7         return $this->get($name."_TITLE");
8     }
9
10    public function getText($name) {
11        return $this->get($name."_TEXT");
12    }
13 }
```

Рисунок 43 – Код `complexmessages_class`

Второй класс `email_class.php` будет не абстрактным, в отличие от предыдущих классов, т.к. здесь будут создаваться экземпляры класса. Данный класс предназначен соответственно для вызова email сообщений. Код представлен на рисунке 44.

```
1 <?php
2 require_once "complexmessage_class.php";
3
4 class Email extends ComplexMessage {
5
6     public function __construct() {
7         parent::__construct("emails");
8     }
9
10 }
```

Рисунок 44 – Код файла `email_class.php`

Для отправки email-писем создаем класс `mail_class.php`. Он использует предыдущие классы, класс с конфигурацией и с шаблонами сообщений. Код файла представлен на рисунке 45.

Далее создаем классы для работы с данными таблиц.

Сначала создаем общий абстрактный класс общий для всех таблиц под названием `global_class.php`, где будут прописаны все базовые методы для общения с данными таблиц. Например вывод определенной таблицы, сортировка данных таблиц и так далее. Фрагмент кода данного файла будет

представлен на рисунке 46. Полный листинг кода представлен в приложении Б.

```
<?php
require_once "config_class.php";
require_once "email_class.php";

class Mail {

    private $config;
    private $email;

    public function __construct() {
        $this->config = new Config();
        $this->email = new Email();
    }

    public function send($to, $data, $template, $from = "") {
        $data["sitename"] = $this->config->sitename;
        if ($from == "") $from = $this->config->admemail;
        $subject = $this->email->getTitle($template);
        $message = $this->email->getText($template);
        $headers = "From: $from\r\nReply-To: $from\r\nContent-type: text/html; charset=utf-8\r\n";
        foreach ($data as $key => $value) {
            $subject = str_replace("{$key}", $value, $subject);
            $message = str_replace("{$key}", $value, $message);
        }
        $subject = '?utf-8?B?' . base64_encode($subject) . '?=';
        return mail($to, $subject, $message, $headers);
    }
}
```

Рисунок 45 – Файл mail\_class.php

```
index.php config_class.php database_class.php format_class.php
1 <?php
2 require_once "database_class.php";
3 require_once "config_class.php";
4 require_once "check_class.php";
5 require_once "url_class.php";
6 require_once "systemmessage_class.php";
7
8 abstract class GlobalClass {
9
10     protected $db;
11     protected $table_name;
12     protected $format;
13     protected $config;
14     protected $check;
15     protected $url;
16 }
```

Рисунок 46 – Фрагмент кода класса global\_class.php

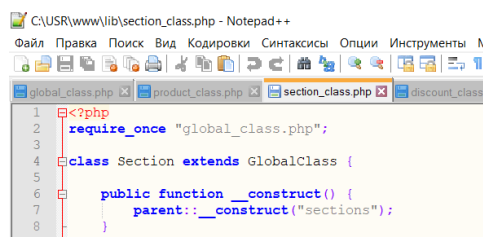
Далее создадим класс product\_class.php, который будет непосредственно заниматься работой с таблицей товаров. Фрагмент кода представлен на рисунке 47, полный листинг файла можно увидеть в приложении Б.

```
global_class.php product_class.php
1 <?php
2 require_once "global_class.php";
3
4 class Product extends GlobalClass {
5
6     public function __construct() {
7         parent::__construct("products");
8     }
9
10    public function getAllData($count) {
11        return $this->transform($this->getAll("date", false, $count));
12    }
13
14    public function getAllTable() {
15        return $this->getAll("id");
16    }
17 }
```

Рисунок 47 – Класс product\_class.php

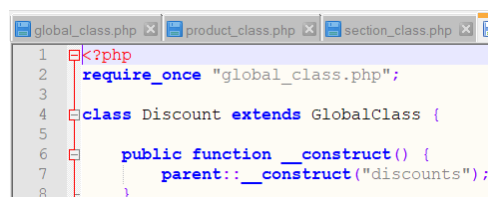
Далее создаем классы `sections_class.php`, для работы с разделами наших товаров, `discount_class.php` для работы со скидками, `order_class.php` для работы с заказами.

Данные классы создаются отдельно и при вызове в других классах используются только результаты. Благодаря этому, пользователь никак не может повлиять на результаты запросов. Фрагменты кода данных файлов представлены на рисунках 48-50, полный линстинг данных файлов можно посмотреть в приложении Б.



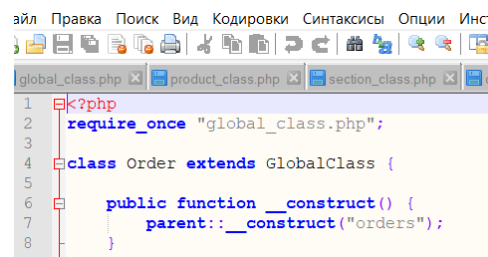
```
1 <?php
2 require_once "global_class.php";
3
4 class Section extends GlobalClass {
5
6     public function __construct() {
7         parent::__construct("sections");
8     }
9 }
```

Рисунок 48 – Класс `sections_class.php`



```
1 <?php
2 require_once "global_class.php";
3
4 class Discount extends GlobalClass {
5
6     public function __construct() {
7         parent::__construct("discounts");
8     }
9 }
```

Рисунок 49 – Класс `discount_class.php`

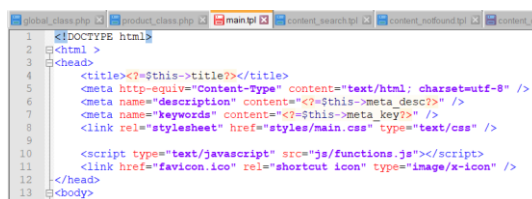


```
1 <?php
2 require_once "global_class.php";
3
4 class Order extends GlobalClass {
5
6     public function __construct() {
7         parent::__construct("orders");
8     }
9 }
```

Рисунок 50 – Класс `order_class.php`

Далее создаем tpl-файлы, которые содержат в себе набор шаблонов, которые будут использоваться при генерации страниц сайта. Они содержат в

себе html-код, который далее будет выводиться на страницу, а так же он содержит в себе вставки php-кода, в которые потом будут выведены значения конкретных переменных. Для начала создаем файл main.tpl, который содержит в себе шаблон главной страницы. Фрагмент кода файла представлен на рисунке 51, полный линстинг можно посмотреть в приложении Б.



```
1 <?DOCTYPE html>
2 <html >
3 <head>
4 <title><?=$this->title?></title>
5 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
6 <meta name="description" content="<?=$this->meta_desc?>" />
7 <meta name="keywords" content="<?=$this->meta_key?>" />
8 <link rel="stylesheet" href="styles/main.css" type="text/css" />
9
10 <script type="text/javascript" src="js/functions.js"></script>
11 <link href="favicon.ico" rel="shortcut icon" type="image/x-icon" />
12 </head>
13 <body>
```

Рисунок 51 – Класс main.tpl

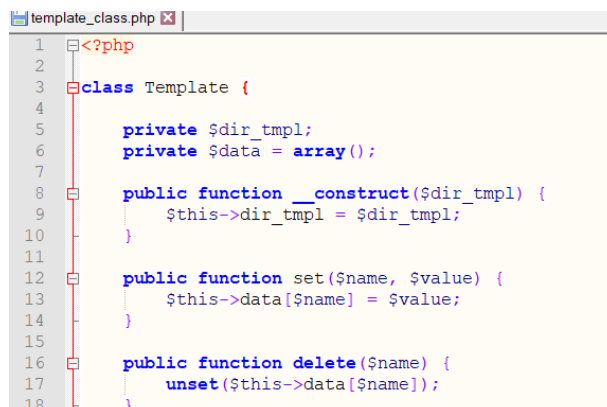
Такие же файлы мы создаем для каждого класса, т.к. классы создаются для каждой страницы, для каждой части. Для корректной работы сайта по аналогии создаем следующие tpl файлы:

- content\_search.tpl – для результатов запросов, которые пользователь вводит в строку поиска;
- content\_notfound.tpl – для вывода сообщения об ошибке при отправке email;
- content\_delivery.tpl – страница оплаты и доставки;
- content\_cart.tpl – страница корзины;
- table\_products\_top.tpl – сортировка;
- table\_products.tpl – товары на главной странице;
- content\_message.tpl – создание сообщения;
- content\_order.tpl – оформление заказа;
- content\_order.tpl – оформление заказа.

Листинг данных файлов находится в приложении Б.

Далее создаем шаблонизатор template\_class.php. Это класс, который отвечает за хранение элементов шаблона. Здесь хранится набор данных,

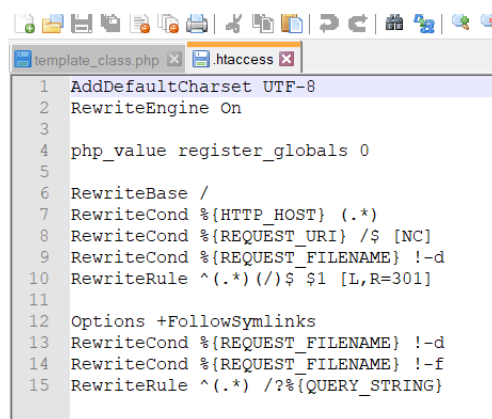
которую хранит какой-нибудь класс, для будущей подстановки значений. Фрагмент кода файла представлен на рисунке 52, полный линстинг можно посмотреть в приложении Б.



```
1 <?php
2
3 class Template {
4
5     private $dir_tmpl;
6     private $data = array();
7
8     public function __construct($dir_tmpl) {
9         $this->dir_tmpl = $dir_tmpl;
10    }
11
12    public function set($name, $value) {
13        $this->data[$name] = $value;
14    }
15
16    public function delete($name) {
17        unset($this->data[$name]);
18    }
19 }
```

Рисунок 52 – Фрагмент класса template\_class.tpl

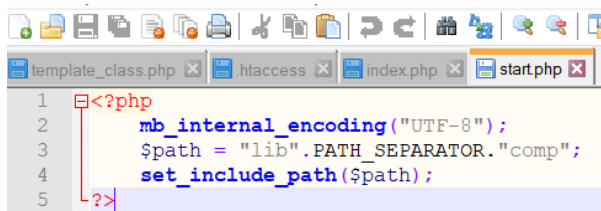
Также, чтобы повысить безопасность данных на сайте, создадим файл .htaccess, который содержит в себе кодировку, функцию удаления следа на конце url сайта и функцию, при которой запросы к несуществующим директориям и файлам автоматически переносят пользователя на главную страницу. На рисунке 53 представлен код данного файла.



```
1 AddDefaultCharset UTF-8
2 RewriteEngine On
3
4 php_value register_globals 0
5
6 RewriteBase /
7 RewriteCond %{HTTP_HOST} (.*)
8 RewriteCond %{REQUEST_URI} /$ [NC]
9 RewriteCond %{REQUEST_FILENAME} !-d
10 RewriteRule ^(.*) (/) $ $1 [L,R=301]
11
12 Options +FollowSymlinks
13 RewriteCond %{REQUEST_FILENAME} !-d
14 RewriteCond %{REQUEST_FILENAME} !-f
15 RewriteRule ^(.*) /?%{QUERY_STRING}
```

Рисунок 53 – Код файла .htaccess

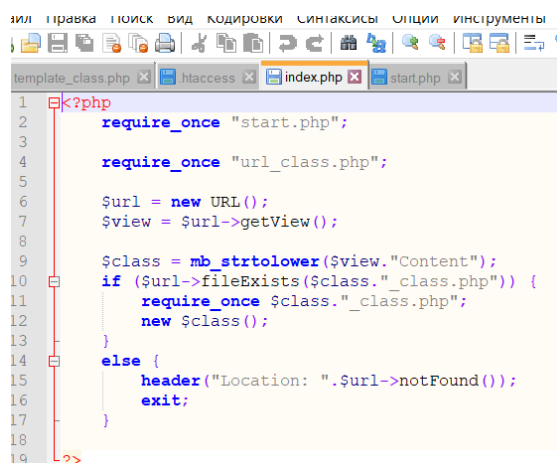
Далее создается файл `start.php`, запускается на старте, содержит в себе кодировку для корректной работы сайта, а так же указывает пути к служебным файлам. На рисунке 54 представлен код данного файла.



```
1 <?php
2     mb_internal_encoding("UTF-8");
3     $path = "lib".PATH_SEPARATOR."comp";
4     set_include_path($path);
5 ?>
```

Рисунок 54 – Код файла `start.php`

Далее создается файл `index.php`. Он подключается к файл `start.php`, подключается к директориям и файлам, а также содержит метод определения класса , чтобы далее создать объект данного класса и вывести его на страницу, а также проверку класса на существование. На рисунке 55 представлен код данного файла.



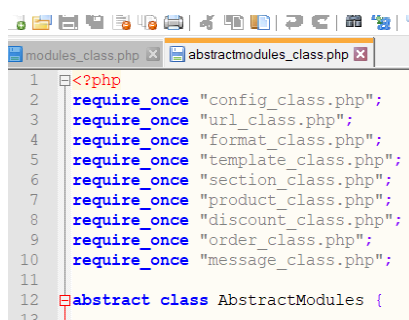
```
1 <?php
2     require_once "start.php";
3
4     require_once "url_class.php";
5
6     $url = new URL();
7     $view = $url->getView();
8
9     $class = mb_strtolower($view."Content");
10    if ($url->fileExists($class."_class.php")) {
11        require_once $class."_class.php";
12        new $class();
13    }
14    else {
15        header("Location: ".$url->notFound());
16        exit;
17    }
18
19 ?>
```

Рисунок 55 – Код файла `index.php`

Далее создается класс `abstractmodules_class.php`, является глобальным классом для всех страниц. Содержит в себе множество библиотек, потому что взаимодействует со всеми классами. Вызывает метод, который позволяет использовать безопасные данные, также содержит путь к `tpl` файлам. Описывает элементы, которые являются общими для всех страниц. На



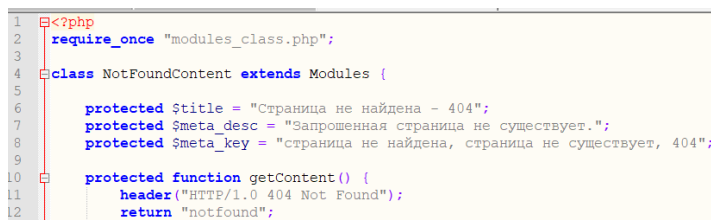
рисунке 56 представлен фрагмент кода данного файла. Полный листинг файла находится в приложении Б.



```
1 <?php
2 require_once "config_class.php";
3 require_once "url_class.php";
4 require_once "format_class.php";
5 require_once "template_class.php";
6 require_once "section_class.php";
7 require_once "product_class.php";
8 require_once "discount_class.php";
9 require_once "order_class.php";
10 require_once "message_class.php";
11
12 abstract class AbstractModules {
13
```

Рисунок 56 – Код файла abstractmodules\_class.php

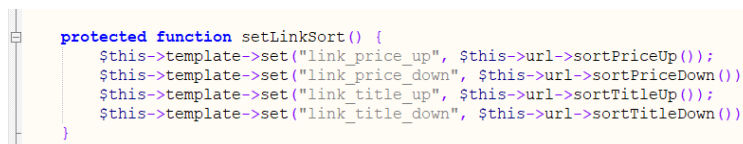
Далее создается файл notfoundcontent\_class.php для вывода страницы с ошибкой 404. Содержит в себе заголовок, который сообщает поисковым системам, что это страницы ошибки, а не спамирование. На данную страницу отправка происходит не только при написании неправильного адреса, но и при некорректном вводе запроса. На рисунке 57 представлен код данного файла. Так же по этому файлу создается tpl файл.



```
1 <?php
2 require_once "modules_class.php";
3
4 class NotFoundContent extends Modules {
5
6     protected $title = "Страница не найдена - 404";
7     protected $meta_desc = "Запрошенная страница не существует.";
8     protected $meta_key = "страница не найдена, страница не существует, 404";
9
10     protected function getContent() {
11         header("HTTP/1.0 404 Not Found");
12         return "notfound";
13     }
14 }
```

Рисунок 57 – Код файла notfoundcontent\_class.php

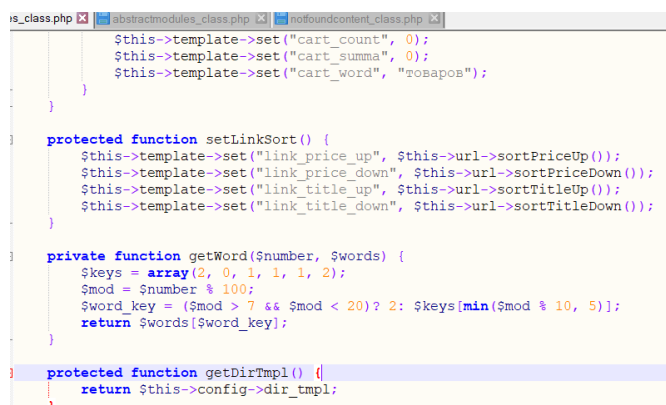
Далее в файле modules\_class.php добавляем функцию сортировки. Данная функция представлена на рисунке 58.



```
protected function setLinkSort() {
    $this->template->set("link_price_up", $this->url->sortPriceUp());
    $this->template->set("link_price_down", $this->url->sortPriceDown());
    $this->template->set("link_title_up", $this->url->sortTitleUp());
    $this->template->set("link_title_down", $this->url->sortTitleDown());
}
```

Рисунок 58 – Добавление сортировки в класс modules\_class.php

Далее организуем вывод по разделам. Для этого создаем файл `sectioncontent_class.php`. Происходит инициализация разделов по их уникальному `id`, далее происходит проверка корректности `id`. А так же содержит мета-теги для страниц разделов. Код представлен на рисунке 59.



```

is_class.php x abstractmodules_class.php x notfoundcontent_class.php x
    $this->template->set("cart_count", 0);
    $this->template->set("cart_summa", 0);
    $this->template->set("cart_word", "товаров");
}

protected function setLinkSort() {
    $this->template->set("link_price_up", $this->url->sortPriceUp());
    $this->template->set("link_price_down", $this->url->sortPriceDown());
    $this->template->set("link_title_up", $this->url->sortTitleUp());
    $this->template->set("link_title_down", $this->url->sortTitleDown());
}

private function getWord($number, $words) {
    $keys = array(2, 0, 1, 1, 1, 2);
    $mod = $number % 100;
    $word_key = ($mod > 7 && $mod < 20)? 2: $keys[min($mod % 10, 5)];
    return $words[$word_key];
}

protected function getDirTpl() {
    return $this->config->dir_tmpl;
}

```

Рисунок 59 –Код файла `sectioncontent_class.php`

Далее вывод страницы с описанием товара. Для этого создаем файл `productcontent_class.php`. Также в этом же файле делаем вывод горизонтальной навигации. Далее в файле содержится метод вывода блока «с этим товаром также заказывают». Так же в класс `product_class.php` добавляем функцию подбора товаров. Линстинг `product_class.php` находится в приложении б. В классе конфигурации мы дописываем код, а именно вписывает константу `count_others`, которая указывает, сколько товаров будет выводиться в данном блоке. Линстинг файла конфигурации также находится в приложении Б. Код файла представлен на рисунке 60.



```

<?php
require_once "modules_class.php";

class ProductContent extends Modules {

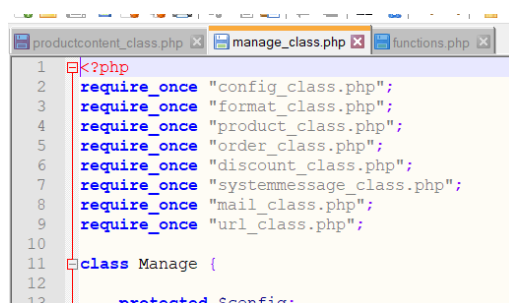
    protected function getContent() {
        $product_info = $this->product->get($this->data["id"], $this->section->getTableName());
        if (!$product_info) return $this->notFound();
        $this->title = $product_info["title"];
        $this->meta_desc = "Описание и покупка товара ".$product_info["title"];
        $this->meta_key = mb_strtolower("описание товара ".$product_info["title"].", купить товар ".$product_info["title"]);

        $this->template->set("link_section", $this->url->section($product_info["section_id"]));
        $this->template->set("product", $product_info);
        $this->template->set("products", $this->product->getOthers($product_info, $this->config->count_others));
        return "product";
    }
}

```

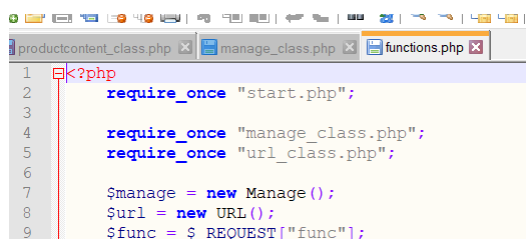
Рисунок 60 –Код файла `productcontent_class.php`

Далее организуем добавление товаров в корзину. Для этого нам нужно создать файл `manage_class.php`, омон является управляющим класс, в котором в сессии содержатся все данные, которые клиент передает на сайт при заказе товара. И файл `functions.php`, в котором организовано добавление и удаление товаров из корзины. Фрагменты кода данных файлов представлены на рисунках 61 и 62. Листинг файлов можно увидеть в приложении Б.



```
1 <?php
2 require_once "config_class.php";
3 require_once "format_class.php";
4 require_once "product_class.php";
5 require_once "order_class.php";
6 require_once "discount_class.php";
7 require_once "systemmessage_class.php";
8 require_once "mail_class.php";
9 require_once "url_class.php";
10
11 class Manage {
12
13     protected $config;
```

Рисунок 61 –Код файла `manage_class.php`



```
1 <?php
2 require_once "start.php";
3
4 require_once "manage_class.php";
5 require_once "url_class.php";
6
7 $manage = new Manage();
8 $url = new URL();
9 $func = $_REQUEST["func"];
```

Рисунок 62 –Код файла `functions.php`

Далее создаем файл `cartcontent_class.php`, который отвечает за вывод содержимого нашей корзины. Сначала из сессии мы получаем массив с выбранными товарами, также здесь идет подсчет количества товара, за счет определения уникальных `id` и их поиска, программа находит одинаковые `id` и суммирует их. А также суммируется общая цена по полю `price`. Далее, в этих файлах организуется пересчет товаров при увеличении количества товара, и пересчет товаров с учетом скидки. Код файла представлен на рисунке 63. Полный листинг файла находится в приложении Б.

```

1 <?php
2 require_once "modules_class.php";
3
4 class CartContent extends Modules {
5
6     protected $title = "Корзина";
7     protected $meta_desc = "Содержимое корзины.";
8     protected $meta_key = "корзина, содержимое корзины";
9
10    protected function getContent() {
11        $cart = array();
12        $summa = 0;
13        if ($SESSION["cart"]) {
14            $ids = explode(",", $SESSION["cart"]);
15            $products = $this->product->getAllOnIds($ids);
16            $result = array();
17            for ($i = 0; $i < count($products); $i++) {
18                $result[$products[$i]["id"]] = $products[$i];
19            }
20            $summa = array_sum($result);
21        }
22    }
23 }

```

Рисунок 63 –Код файла cartcontent\_class.php

Для организации пересчета с учетом скидки создаем файл discount\_class.php. Данный файл использует данные из поля code, таблицы mysql, где указан код и размер скидки. Так же, класс содержит в себе сообщения об ошибках, если код скидки введен неверно. Код файла представлен на рисунке 64.

```

1 <?php
2 require_once "global_class.php";
3
4 class Discount extends GlobalClass {
5
6     public function __construct() {
7         parent::__construct("discounts");
8     }
9
10    public function getOnCode($code) {
11        return $this->getOnField("code", $code);
12    }
13
14    public function getTableData($count, $offset) {
15        return $this->transform($this->getAll("id", true, $count, $offset));
16    }
17
18    protected function transformElement($discount) {
19        $discount["link_admin_edit"] = $this->url->adminEditDiscount($discount["id"]);
20        $discount["link_admin_delete"] = $this->url->adminDeleteDiscount($discount["id"]);
21        return $discount;
22    }
23
24    public function getValueOnCode($code) {
25        return $this->getField("code", $code, "value");
26    }
27
28    protected function checkData($data) {
29        if (!$this->check->title($data["code"])) return "ERROR_CODE";
30        if (!$this->check->value($data["value"])) return "ERROR_VALUE";
31        return true;
32    }
33 }

```

Рисунок 64 –Код файла cartcontent\_class.php

Далее создаем файл ordercontent\_class, который содержит в себе страницу оформления заказа и сохраняет в сессии данные, которые ввел пользователь. Код файла представлен на рисунке 65.



```

<?php
require_once "modules_class.php";

class OrderContent extends Modules {

    protected $title = "Оформление заказа";
    protected $meta_desc = "Оформление заказа на покупку.";
    protected $meta_key = "заказ, оформление заказа";

    protected function getContent() {
        $this->template->set("message", $this->message());
        $this->template->set("name", $_SESSION["name"]);
        $this->template->set("phone", $_SESSION["phone"]);
        $this->template->set("email", $_SESSION["email"]);
        $this->template->set("delivery", $_SESSION["delivery"]);
        $this->template->set("address", $_SESSION["address"]);
        $this->template->set("notice", $_SESSION["notice"]);
        return "order";
    }
}

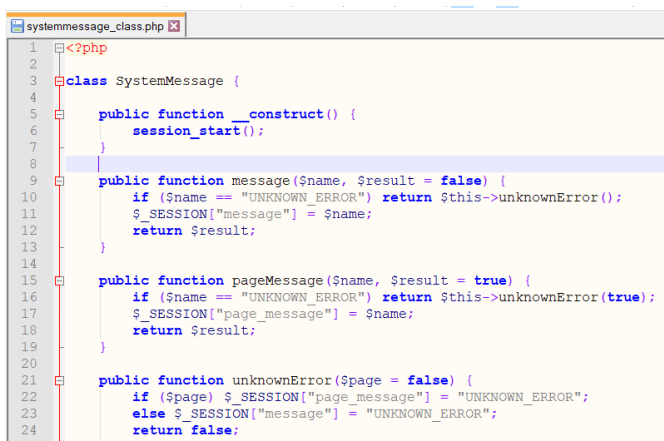
```

Рисунок 65 –Код файла ordercontent\_class

Приступим к обработке формы оформления заказа. Целью данной формы является перенос данных введенным пользователем в таблицу orders. Далее организуем перенос введенных данных в таблицу бд. Для этого в файле database\_class.php создаем метод, который является не селектом, а позволяет либо обновить, удалить или добавить запись. Если запрос не на добавление записи, но прошел удачно, значит был выполнен запрос на обновление или удаление записи. Иначе, возвращается id вставляемой записи. С кодом данного метода можно ознакомиться в листинге файла в приложении б. Далее в global\_class.php создаем функцию добавления записи и проверку данных. Вызывается эта функция в классе order\_class.php. Данный код можно увидеть в листинге файла global\_class.php в приложении б. После в классе function.php пишем функцию добавления, в класса manage\_class.php создается и передается массив. Листинг файлов можно увидеть в приложении Б.

Далее организуем страницу, где пользователю сообщается, что все данные введены верно и с ним свяжется администратор или наоборот, сообщается об ошибках. Для данных сообщений создаем файл systemmessages\_class.php, там мы обращаемся к уже созданным файлам с сообщениями (а именно message.ini, page\_message.ini). Далее в файле global\_class.php в проверку данных добавляем, для вывода на ту же страницу сообщения об ошибке, в файле modules\_class.php указываем, какой текст выводится при конкретной ошибке. Pagemessage\_class.php отвечает за вывод

мелких ошибок на той же странице, а класс `messagecontent_class.php` сообщает о глобальных сообщениях, которые переносят на другую страницу. Код файла `systemmessage_class.php` представлен на рисунке 66, код файла `pagemessage_class.php` представлен на рисунке 67, код файла `messagecontent_class.php` представлен на рисунке 68, с листингом остальных файлов можно ознакомиться в приложении Б.

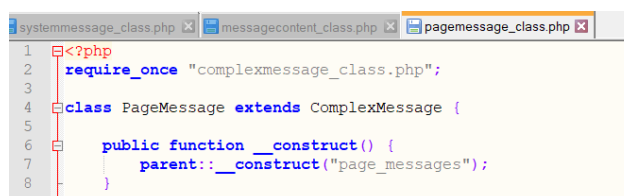


```

1 <?php
2
3 class SystemMessage {
4
5     public function __construct() {
6         session_start();
7     }
8
9     public function message($name, $result = false) {
10        if ($name == "UNKNOWN_ERROR") return $this->unknownError();
11        $_SESSION["message"] = $name;
12        return $result;
13    }
14
15    public function pageMessage($name, $result = true) {
16        if ($name == "UNKNOWN_ERROR") return $this->unknownError(true);
17        $_SESSION["page_message"] = $name;
18        return $result;
19    }
20
21    public function unknownError($page = false) {
22        if ($page) $_SESSION["page_message"] = "UNKNOWN_ERROR";
23        else $_SESSION["message"] = "UNKNOWN_ERROR";
24        return false;
25    }
26

```

Рисунок 66 – Код файла `systemmessages_class.php`

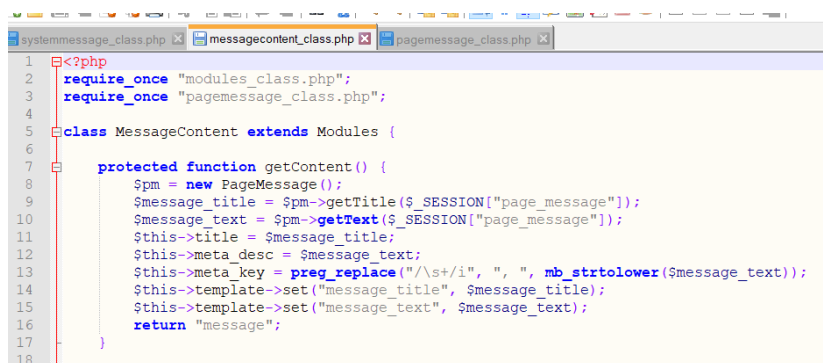


```

1 <?php
2 require_once "complexmessage_class.php";
3
4 class PageMessage extends ComplexMessage {
5
6     public function __construct() {
7         parent::__construct("page_messages");
8     }
9

```

Рисунок 67 – Код файла `pagemessage_class.php`



```

1 <?php
2 require_once "modules_class.php";
3 require_once "pagemessage_class.php";
4
5 class MessageContent extends Modules {
6
7     protected function getContent() {
8         $pm = new PageMessage();
9         $message_title = $pm->getTitle($_SESSION["page_message"]);
10        $message_text = $pm->getText($_SESSION["page_message"]);
11        $this->title = $message_title;
12        $this->meta_desc = $message_text;
13        $this->meta_key = preg_replace("/\s+/i", " ", mb_strtolower($message_text));
14        $this->template->set("message_title", $message_title);
15        $this->template->set("message_text", $message_text);
16        return "message";
17    }
18

```

Рисунок 68 – Код файла `messagecontent_class.php`

Далее организуем вывод результатов поиска. Создаем файл `searchcontent_class.php`. Там запрос пользователя переносится в переменную `q`. В классе `global_class.php` создаем функцию и указываем поля, по которым будет производиться поиск, а также как будет производиться поиск. После данных действий, класс `searchcontent_class.php` выводит результаты поиска и добавляет к ним возможность сортировки результатов. Код файла `searchcontent_class.php` представлен на рисунке 68.



```
1 <?php
2 require_once "modules_class.php";
3
4 class SearchContent extends Modules {
5
6     protected function getContent() {
7         $q = $this->data["q"];
8         $this->title = "Поиск: $q";
9         $this->meta_desc = "Поиск: $q.";
10        $this->meta_key = preg_replace("/\s+/i", " ", mb_strtolower($q));
11
12        $this->setLinkSort();
13        $this->template->set("q", $q);
14        $this->template->set("table_products_title", "Поиск");
15        $this->template->set("products", $this->product->search($q, $this->data["sort"], $this->data["up"]));
16        return "search";
17    }
18 }
```

Рисунок 68 – Код файла `searchcontent_class.php`

Далее делаем оставшиеся страницы, а именно контакты и оплата и доставка. Для этого создаем файл `deliverycontent_class.php`, к нему `content_delivery.tpl`, где находится шаблон вывода содержимого на страницу. Для страницы контакты проделываем те же действия.

Далее создаем admin-панель. Admin-панель – это отдельный сайт, созданный для управления базой данных.

Сначала создается верстка сайта.

Затем переходим к созданию движка. Структура движка будет такая же, как и для основного сайта.

Данная панель будет использовать как свою личную, так и общие директории, которыми пользуется основной сайт. Это те классы, которые отвечают за какой-либо контент. Данные классы представлены на рисунке 69.

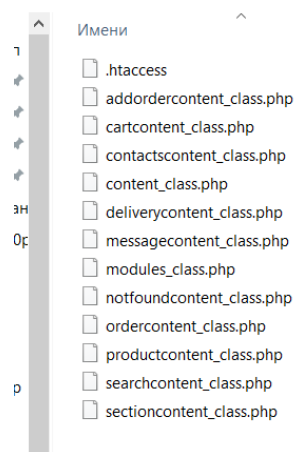


Рисунок 69 – Общие классы для admin-панели и сайта

По аналогии, для admin-панели создаем отдельные tpl файлы для каждой страницы. Все tpl-файлы представлены на рисунке 70.

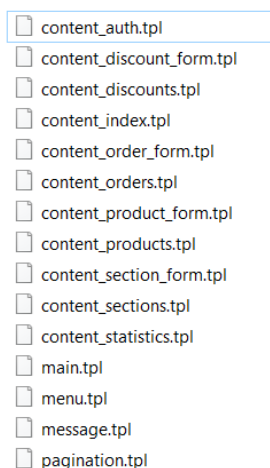


Рисунок 70 – Tpl-шаблоны для панели администратора

Сначала создается главная страница. Листинг главной станицы можно увидеть в приложении Б.

На главной странице будут располагаться ссылки на такие страницы, как главная, товары, заказы, разделы, купоны, статистика, и выход а также таблица со статистикой продаж за последние 7 дней.

Далее приступаем к созданию механизма авторизации администратора. Данный механизм будет организован с помощью сессий. Для этого создаем



новый класс `auth_class.php`. Данный класс будет занимать проверкой правильности логина и пароля. С листингом данного файла можно ознакомиться в приложении Б.

Далее создаем постраничную навигацию для статистики. На странице будет размещаться по 10 заказов. Организацию данной функции можно увидеть файлах `adminmodules_class.php` и `urladmin_class.php` в приложении Б.

Далее приступаем к созданию класса вывода данных. На всех страницах будут присутствовать таблицы, у которых будут изменяться названия столбцов и само наполнение. Для того, чтобы не составлять дублирование кода, создадим вспомогательный класс под названием `adminform_class.php`. Там описывается, как данные будут передаваться на вывод, а также, как они будут себя вести при различных преобразованиях, например при добавлении записей или изменении существующих. С листингом данного файла можно ознакомиться в приложении Б.

Далее приступаем к организации вывода товаров на сайт. Для этого создаем новый файл под названием `adminproductscontent_class.php`, который будет тесно взаимодействовать с классом `adminform_class.php`. Данный класс наследует класс `adminform_class.php`. В этих двух классах не происходит преобразование данных, здесь лишь даются инструкции по выводу этих данных. Листинг данного файла можно увидеть в приложении Б.

Далее занимаемся организацией формы добавления нового товара на сайт. Для начала, в данной форме будет название формы (в данном случае форма добавления товара). Выпадающий список, для выбора раздела товара, поля для ввода всей необходимой информации описания товара и кнопка добавления. Для вывода разделов в файле `adminproductscontent_class.php` прописывается функция. Затем переходим в `functions.php`, там добавим функцию добавления товара и функцию, которая после отправки формы добавления товара, перенаправит администратора на страницу товаров. Так как формы добавления и редактирования товара будут одинаковыми, следует создать общий метод для преобразований, который будет возвращать массив

с данными. Данный метод напишем в файле `manageadmin_class.php`. Здесь же прописываются проверки, например проверка расширения изображения. Также в файле `config_class.php` указываем максимальный размер изображения, который можно загрузить. С кодом данных файлов можно ознакомиться в приложении Б. Для сообщения об ошибках в файл `message.ini` добавляем соответствующие сообщения.

Далее приступаем к редактированию товаров. Для этого в `function.php` добавляем блок функции `edit`. Саму функцию прописываем в `manageadmin_class.php`. Метод `edit` прописываем в `global_class.php`. Так же в `message.ini` прописываем сообщения об ошибках.

Далее реализуем функцию удаления. Так же в `functions.php` добавляем блок `delete`. Добавляем в `manageadmin_class.php` функцию. Удаление записи происходит только по `id` товара, только оно передается по функциям и страницам. После удаления возвращается сообщение об успешном удалении или об ошибке. Метод прописывается в `global_class.php`. Также добавляем сообщения в `message.ini`.

Далее организуем управление разделами. Создаем новый класс `adminsectionscontent_class.php`. Там находится описание страницы разделов. В файле `sections_class.php` организуем сортировки по названию и цене. Также дублируем функции добавления, редактирования и удаления. Также организуется проверка входных данных и сообщений об уведомлениях совершенных действий. С листингом данных файлов можно ознакомиться в приложении Б.

Аналогичным образом организуется добавление, редактирование и удаление заказов, а также сообщений о результате данных действий. Мы используем следующие файлы `manageadmin_class.php`, `order_class.php` и `global_class.php`, `messages.ini`. Листинги используемых файлов находятся в приложении Б.

Далее занимаемся выводом формы редактирования заказа. Основную сложность составляет перенос данных на форму и отправка их для выведения

на странице. Реализация представлена в файле `adminorderscontent_class.php` в приложении Б.

Далее занимаемся редактированием позиций заказа. Блок редактирования организуем с помощью `java`-скрипта. Листинг данного файла можно увидеть в файле `fnctions.js` в приложении Б.

Далее занимаемся управлением купонами на скидки, а именно добавлением, редактированием и удалением. Заполняем класс `admindiscountscontent_class.php`. В классе `discount_class.php` прописываем функцию, как и для предыдущих разделов, для получения таблицы данных. В `manageadmin_class.php` прописываем функции для их редактирования, добавления и удаления. В файле `discount_class.php` создаем функцию проверки данных, а также соответствующие сообщения в `messages.ini`.

Далее приступаем к разделам «Статистика» и «Главная». Там будут выводиться данные, а именно количество заказов, сумму счетов, доход и количество купленных товаров, за определенный период. Для этого создаем класс `statistics_class.php`, где подключаем класс `order_class.php` для инициализации поля заказов, чтобы получить данные для администратора. Организуем форму, куда будут выводиться данные в файле `adminstatistics_class.php`. Листинг файлов можно увидеть в приложении Б.

На этом основная разработка сайта и панели администратора для компании «Велес Монтаж» заканчивается.

Далее приступим к дополнениям данного сайта.

На странице с контактами разместим интерактивную карту, чтобы клиентам было легче найти магазин. Делается это достаточно просто, на сайте Яндекс.карты отмечаем компанию «Велес Монтаж», прописываем в поле указатели «Мы находимся здесь! Ждем вас!», и получаем автоматически сформированный скрипт, который добавляет в `tpl`-файл страницы с контактами.

Для обновления базы данных будет производиться выгрузка в формате `csv` с сервера базы данных товаров. Далее через `cron` на сервере хостинг-

провайдера будет настроен автоматический запуск скрипта для обновления данных на нашем сайте. За этот отвечает скрипт `update.php`, листинг которого можно увидеть в приложении Б.

Так же, периодически будут проводиться через социальную сеть Вконтакте конкурсы с целью привлечения клиентов. Для получения списка пользователей, которые поделились ссылкой на запись, в удобном формате был разработан скрипт `repost.php`. Листинг данного файла можно увидеть в приложении Б.

Для проведения автоматической рассылки сообщений e-mail по требованию магазина необходимо делать это по запросу менеджера. Для этого будет создан простой интерфейс отправки списку пользователей.

Однако почтовые клиенты и почтовые сервисы имеют множественную проверку писем на спам. Для того, чтобы письмо от компании было максимально доверенным, создадим скрипт, который будет совершать отправку в цикле по одному пользователю (чтобы исключить множественные отправки), а также использовать корпоративный сервер Exchange. Поможет нам в этом библиотека PHPmailer. Отвечать за отправку будет скрипт `mailer.php`. Код данного скрипта находится в приложении Б.

### **3.3 Оценка экономической эффективности внедрения информационно-аналитического обеспечения**

Общепринятый инструмент обоснования любого бизнес-проекта – инвестиционный анализ. Для оценки инвестиционной рентабельности ИТ проекта чаще всего применяются так называемые динамические методы, основанные преимущественно на дисконтировании образующихся в ходе реализации проекта денежных потоков. Таким образом, методы инвестиционного анализа позволяют оценить экономические параметры внедрения информационных систем по аналогии с оценкой любого другого инвестиционного проекта.

PP (Payback Period) – Метод расчета срока окупаемости инвестиций – один из самых простых и широко распространенных в мировой учетно-аналитической практике. Его алгоритм зависит от равномерности распределения прогнозируемых доходов от инвестиций: если доход распределен по годам равномерно, то срок окупаемости рассчитывается делением единовременных затрат на величину годового дохода, обусловленного ими; если прибыль распределена неравномерно, то срок окупаемости рассчитывается прямым подсчетом числа лет, в течение которых инвестиция будет погашена кумулятивным доходом. По сути, PP представляет собой анализ возврата средств исходя из принятых в компании максимальных сроков окупаемости вложений.

Анализируя деятельность действующего сайта и работу над созданием и продвижением нового обеспечения, можно сделать следующие предположительные расчеты и оценки.

Для оценки рассчитываются следующие показатели:

- цена продукта. (Ц);
- дополнительные доходы от реализации программного продукта после размещения web-сайта составили. (Ддоп);
- экономическая эффективность (прибыль) (Э);

– рентабельность проекта (Р).

Цена программного продукта (web-сайта) Ц определяется итоговыми затратами и прибылью, которая, в свою очередь, составляет 30 процентов (условно) от фонда заработной платы.

$$\text{Э (П)} = \text{Ддоп} - \text{Ц} \quad (1.1)$$

$$P = \text{П} / \text{Ц} * 100 \quad (1.2)$$

$$\text{Ток} = \text{Ц} / \text{П} \quad (1.3)$$

Рассмотрим технико-экономическое обоснование проекта на примере предприятия ООО «Велес Монтаж», на котором внедрялся web-сайт.

При анализе экономической эффективности в первую очередь необходимо рассчитать стоимость разработки web-сайта. Используя результаты этих расчетов, следует вычислить экономическую эффективность результатов деятельности ООО «Велес Монтаж» по реализации программного обеспечения после опубликования web-сайта.

График выполнения работ представлен в таблице 1.1

Таблица 1.1 – График выполнения работ

№ п/п	Наименование работ	Исполнитель	Длительность работы в апреле – мае 2018 года	
			В днях	В часах
1	2	3	4	5
1	Разработка технического задания	Якимова М.Е.	1	8
2	Планирование сайта	Якимова М.Е.	2	16
3	Рабочее проектирование сайта	Якимова М.Е.	10	80
4	Отладка и тестирование	Якимова М.Е.	7	56
5	Обобщение и оценка результата	Якимова М.Е.	3	24
	Итого	Х	23	184

Таким образом, на разработку сайта для ООО «Велес Монтаж» было затрачено 23 дня в апреле – мае 2018 года, или 184 человеко-часа (при восьмичасовом рабочем дне).

При расчете стоимости (составлении сметы затрат) разработки сайта учитываются следующие виды расходов:

- стоимость материалов и покупных изделий;
- основная заработная плата;
- дополнительная заработная плата;
- страховые взносы;
- накладные расходы;
- затраты на машинное время (затраты на электроэнергию).

Перечень затрат на материалы и покупные изделия приведен в таблице 1.2.

Таблица 1.2 – Затраты на материалы и покупные изделия

№ п/п	Наименование	Единица измерения	Количество	Цена за единицу, руб.	Стоимость, руб.
1	2	3	4	5	6
1	Плата за интернет	Х	Х	Х	1000
2	Канцтовары	шт.	2	50	100
3	Бумага формата	упаковка	1	121	121
4	Компакт-диск с программами	шт.	1	129	129
5	Итого	Х	Х	Х	1350
6	Транспортные расходы (10% от п.6)	Х	Х	Х	135
7	Итого	Х	Х	Х	1485

Транспортные расходы учитываются в объеме 10% от суммы затрат на материалы и покупные изделия, что составляет 135 руб. Таким образом, затраты на материалы и покупные изделия равны

$$З_{\text{м}} = 1350 + 1350 \cdot 10 / 100 = 1485 \text{ руб.} \quad (1.4)$$

Результаты расчета фонда платы за создание сайта представлены в таблице 1.3.

Таблица 1.3 – Расчет фонда заработной платы

№ п/п	Компания	Количество рабочих дней	Размер дневной оплаты, руб.	Заработная плата в месяц, руб.
1	2	3	4	5
1	Основная заработная плата	30	1083	32500
2	Итого фонд заработной платы	X	X	32500

Следовательно, разработчику web-сайта всего начислено 32500 руб.

Как следует из данных таблицы 1.1, на разработку и последующую отладку сайта для ООО «Велес Монтаж» потребовалось 23 рабочих дня (Дн). В среднем с учетом перерывов программист работает за компьютером 6 часов в день. Себестоимость одного кВт/ч электроэнергии (С1кВт/ч) для организаций составляет 2 рубля 88 копеек. При проведении расчетов в проекте необходимо в расчеты брать существующие на дату расчета тарифы. Суммарная мощность энергопотребителей для АРМ программиста складывается из мощности, потребляемой системным блоком персонального компьютера, монитором, принтером и другим периферийным оборудованием, которая составляет 1,2 кВт. Следовательно, за 6 часов работы программиста суммарное энергопотребление за день составит:

$$P = 1,2 * 6 = 7,2 \text{ кВт/ч} \quad (1.5)$$

Машинное время разработки web-сайта, составит:

$$Z_{\text{маш}} = P * \text{Дн} * C1_{\text{кВт/ч}} = 7,2 \text{ кВт/ч} * 23 * 2,88 \text{ руб./кВт/ч} = 476,93 \text{ руб.} \quad (1.6)$$



Затраты на машинное время учитываются как затраты на электроэнергию.

В результате выше произведенных расчетов мы получили итоговые затраты на разработку web-сайта (Таблице 1.4.).

Таблица 1.4 – Затраты на разработку веб-сайта.

№ п/п	Наименование статей расходов	Сумма, руб.
1	2	3
1	Стоимость материалов и покупных изделий	1485,00
2	Основная заработная плата	32 500,00
3	Затраты на машинное время (затраты на электроэнергию)	476,93
4	Итого	34 461,93

Цена программного продукта (web-сайта) Ц определяется итоговыми затратами и прибылью, которая, в свою очередь, составляет 30 процентов (условно) от фонда заработной платы:

$$Ц = 34461,93 + 32500 * 30/100 = 44211,93 \text{ руб.} \quad (1.6)$$

Оценка экономической эффективности.

Одним из основных методов определения экономической эффективности служит метод сравнения результатов работы до и после проведения мероприятия. Расчет экономической эффективности по этому методу проведен на основе данных о количестве принятых и обработанных заказов на программное обеспечение до и после размещения нового сайта ООО «Велес Монтаж». Усредненные данные о реализации программного обеспечения приведены в таблице 1.5.

Таблица 1.5 – Усредненные данные о реализации программного обеспечения в сравнении со старым обеспечением.

№ п/п	Наименование товара	Стоимость, руб.	Количество принятых заказов (ед.)		Доходы от реализации программного обеспечения (руб)	
			До	После	До	После
1	2	3	4	5	6	7
1	Daikin FTYN20L/RYN20L	41932	1	2	41932	83864
2	Neoclima NHL-NHL-220L	2080	1	3	2080	6240
3	Вытяжка кухонная Aralia 250 Black	5590	1	3	5590	11180
4	Итого	X	X	X	49602	100884

Из приведенных данных можно сделать выводы о том, что дополнительные доходы от реализации программного продукта в результате размещения сайта ООО «Велес Монтаж» составили:

$$\text{Ддоп} = 100884 - 49602 = 51282 \text{ руб.} \quad (1.7)$$

Расходы на разработку сайта (цена программного продукта) составили 44211,93 рублей. Однако, так как данное обеспечение было создано в рамках преддипломной практики, организация понесла минимальные расходы на канцелярию и электроэнергию в объеме 5 тыс. руб, а само обеспечение создано бесплатно.

Даже с расчетом платного создания обеспечения, экономический эффект в 1,5 раза превышает расходы на обработку сайта.

Приведенные данные говорят о заметном экономическом эффекте от будущего размещения сайта ООО «Велес Монтаж», превышающем затраты на разработку и внедрение сайта.

## ЗАКЛЮЧЕНИЕ

В рамках создания выпускной квалификационной работы были применены на практике навыки создания информационно-аналитического обеспечения на примере магазина климатической техники ООО «Велес Монтаж». Данное обеспечение можно использовать для реальной торговой деятельности.

В ходе написания выпускной квалификационной работы, были выполнены следующие задачи:

- изучена и проанализирована деятельность компании;
- изучены методы и инструменты удержания и привлечения клиентов;
- рассмотрены и выявлены недостатки существующего информационно-аналитического обеспечения;
- проанализированы варианты реализации информационно-аналитического обеспечения;
- выбраны программные средства разработки информационно-аналитического обеспечения;
- разработано информационно-аналитическое обеспечение;
- рассчитаны затраты и эффективность информационно-аналитического обеспечения.

Таким образом, в процессе написания выпускной квалификационной работы был получен полезный навык разработки информационно-аналитического обеспечения, который можно использовать в дальнейшей профессиональной деятельности.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Абрютина, М. Анализ финансово-экономической деятельности организации / М. Абрютина. – Москва: Дело и Сервис, 2013. - с.256
2. Аренков, И. Маркетинговые исследования: основы теории и методики / И. Аренков. – Санкт-Петербург: СПбУЭФ, 2016. – 198с.
3. Аткинсон, Л. MySQL. Библиотека профессионала / Л. Аткинсон. – Москва: Вильямс, 2015. – 624 с.
4. Балабанов, И. Электронная коммерция / И. Балабанов. – Москва: Питер, 2015. – 335 с.
5. Бейли, Л. Изучаем PHP и MySQL / Л. Бейли. – Москва: Эксмо, 2013. – 768 с.
6. Белый, С. 55 способов привлечь миллион клиентов / С. Белый. – Москва: Питер, 2013. – 271 с.
7. Борисенко, А. Web-дизайн. Просто как дважды два / А. Борисенко. – Москва: Эксмо, 2015. – 320 с.
8. Веллинг, Л. Разработка Web-приложений с помощью PHP и MySQL / Л. Веллинг. – Москва: «Вильямс», 2014. – 880 с.
9. Вовк, Е. Верстка для начинающих. PageMaker, Photoshop и CorelDraw / Е. Вовк. – Москва: КУДИЦ-Образ, 2017. – 400 с.
10. Гаевский, А. 100% самоучитель. Создание Web-страниц и Web-сайтов. HTML и JavaScript / А. Гаевский. – Москва: Триумф, 2017. – 464 с.
11. Гарретт, Д. Веб-дизайн. Элементы опыта взаимодействия / Д. Гарретт. – Санкт-Петербург: Символ-плюс, 2015. – 192 с.
12. Голицына, О. Информационные технологии / О. Голицына. – Москва: Форум, 2015. – 234 с.
13. Грачев, А. Создаем свой сайт на WordPress: быстро, легко, бесплатно / А. Грачев. – Санкт-Петербург: Питер, 2014. – 281 с.
14. Гроховский, Л. Продвижение порталов и интернет-магазинов / Л. Гроховский. – Москва: Эксмо, 2017. – 224 с.

15. Дакетт, Дж. HTML и CSS. Разработка и дизайн веб-сайтов (+ CD-ROM) / Дж. Дакетт. – Москва: Эксмо, 2013. – 480с.
16. Дакетт, Дж. Основы веб-программирования с использованием HTML, XHTML и CSS / Дж. Дакетт. – Москва: Эксмо, 2015. – 768 с.
17. Дронов, В. HTML 5, CSS 3 и Web 2.0. Разработка современных Web-сайтов / В. Дронов. – Москва: БХВ-Петербург, 2016. – 416 с.
18. Зандстра, М. PHP. Объекты, шаблоны и методики программирования / М. Зандстра. – Москва: Вильямс, 2016. – 576 с.
19. Иванова, П. Веб-сайт как инструмент формирования имиджа организации / П. Иванова, Н. Г. Соколова // Научные труды Sword. – 2013. – №40-1. – 35 с.
20. Костин, С. Самоучитель создания Web-сайтов / С. Костин. – Москва: Триумф, 2013. – 176 с.
21. Кузнецов, М. Объектно-ориентированное программирование на PHP / М. Кузнецов. – Москва: БХВ-Петербург, 2014. - 608 с.
22. Ларин, М. Проблемы совершенствования автоматизации управления в современных условиях / М. Ларин. – Москва: ИТ Пресс, 2014. – 181 с.
23. Макфарланд, Д. Большая книга CSS3: учебное пособие / Д. Макфарланд. – Санкт-Петербург: Питер, 2016. – 608 с.
24. Митчелл, Скотт 5 проектов Web-сайтов от фотоальбома до магазина / Скотт Митчелл. – Москва: ИТ Пресс, 2013. – 224 с.
25. Нечепуренко, Т. Кросс-продажи. Революционные техники продаж / Т. Нечепуренко. – Москва: Омега-Л, 2013. – 176 с.
26. Никсон, Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5 / Р. Никсон. – Москва: Питер, 2016. – 768 с.
27. Роббинс, Д. HTML5, CSS3 и JavaScript. Исчерпывающее руководство: учебное пособие / Д. Роббинс. – Москва: Эксмо, 2014. – 528 с.

28. Салбер, А. Как открыть интернет-магазин / А. Салбер. – Москва: SmartBook, 2014. – 320 с.
29. Сергеев, А. Раскрутка сайтов и основы электронной коммерции. Краткое руководство / А. Сергеев. – Москва: Вильямс, 2016. – 256 с.
30. Сырых, Ю. Современный веб-дизайн. Настольный и мобильный / Ю. Сырых. – Москва: Вильямс, 2014. – 384 с.
31. Тришин, М. Генератор новых клиентов. 99 способов массового привлечения покупателей / М. Тришин. – Москва: Питер, 2013. – 224 с.
32. Фокс, Дж. Как стать волшебником продаж. Правила привлечения и удержания клиентов / Дж. Фокс. – Москва: Альпина Паблишер, 2014. – 164 с.
33. Фрейен, Б. HTML5 и CSS3. Разработка сайтов для любых браузеров и устройств / Б. Фрейен. – Санкт-Петербург: Питер, 2014. – 304 с.
34. Ших, К. Эра Facebook. Как использовать возможности социальных сетей для развития вашего бизнеса / К. Ших. – Москва: БХВ-Петербург, 2016. – 304 с.
35. Шоул, Дж. Первокласный сервис как конкурентное преимущество / Дж. Шоул. – Москва: ООО Альпина Паблишер, 2013. – 256 с.
36. Эванс, Э. Предметно-ориентированное проектирование (DDD). Структуризация сложных программных систем / Э. Эванс. – Москва: Вильямс, 2013 – 448 с.
37. Яргер, Дж. MySQL и mSQL: Базы данных для небольших предприятий и Интернета / Дж. Яргер. – Москва: Символ-Плюс, 2014. – 560 с.
38. SEO. Искусство раскрутки сайтов / Энж Эрик, Стеф. Спенсер, Р. Фишкин, Дж. Стрикчиола – Москва: БХВ-Петербург, 2014. – 668 с.
39. Википедия информационный портал [Электронный ресурс]. – Режим доступа: <http://ru.wikipedia.org/wiki>
40. Конструктор карт Яндекс [Электронный ресурс]: Создание новой карты [сайт] – Режим доступа: <https://yandex.ru/map-constructor/>

41. Общероссийский классификатор видов экономической деятельности [Электронный ресурс]: Изменения 12/2016 ОКВЭД 2, утв. Приказом Росстандарта от 07.10.2016 N 1326-ст. – Режим доступа: <http://regforum.ru/okved/>

42. Google support [Электронный ресурс] – Справка о протоколе HTTPS – Режим доступа: <https://support.google.com/webmasters>

43. LiveInternet интернет порта [Электронный ресурс]: – Новости [сайт] – Режим доступа: <https://www.liveinternet.ru/>

44. Notepad++ Текстовый редактор [Электронный ресурс]: - Перечень плагинов – Режим доступа: <https://notepad-plus-plus.org/>

45. PageSpeed Insights Сервис рекомендаций разработчикам [Электронный ресурс] – Анализ сайта – Режим доступа: <https://developers.google.com/speed/pagespeed/insights/>

46. PHP-Start Курс программирования на языке php [Электронный ресурс] – PHP теория и практика – Режим доступа: <https://php-start.com/course>

47. Ukit [Электронный ресурс] – Конструктор сайтов [сайт] – Режим доступа: <https://ukit.com/ru>

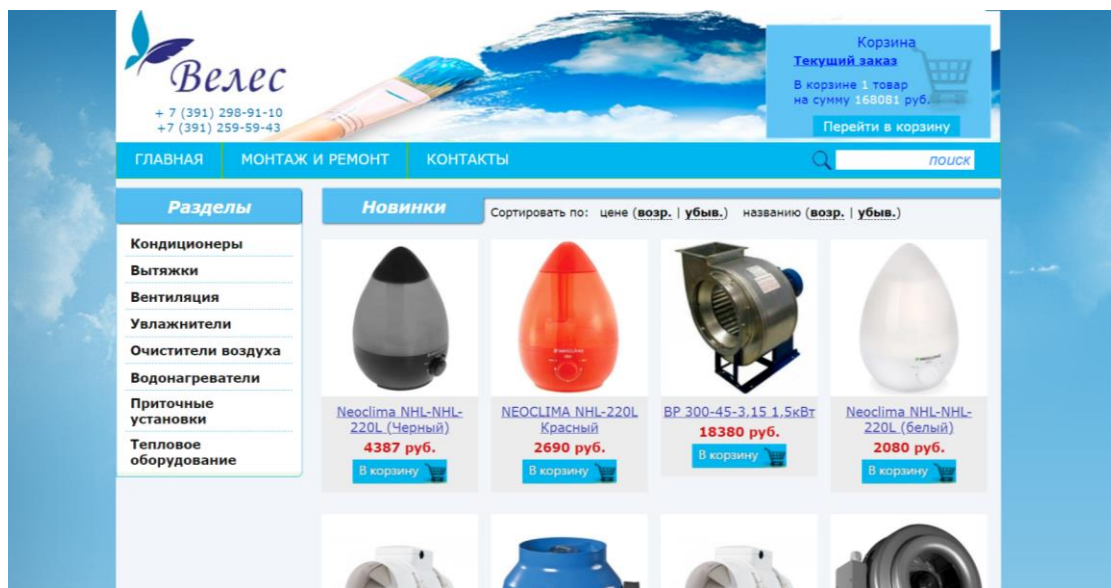
48. Wix [Электронный ресурс] – Конструктор сайтов тарифы [сайт] – Режим доступа: <https://ru.wix.com/>

49. WordPress [Электронный ресурс] – Конструктор сайтов тарифы [сайт] – Режим доступа: <https://ru.wordpress.org/>

50. 2Гис Городской информационный сервис [Электронный ресурс]: Тарифы для компаний [сайт] – Режим доступа: <http://reklama.2gis.ru/krasnoyarsk>

## ПРИЛОЖЕНИЕ А

### Главная страница сайта ООО «Велес Монтаж»



### Страница товара сайта ООО «Велес Монтаж»

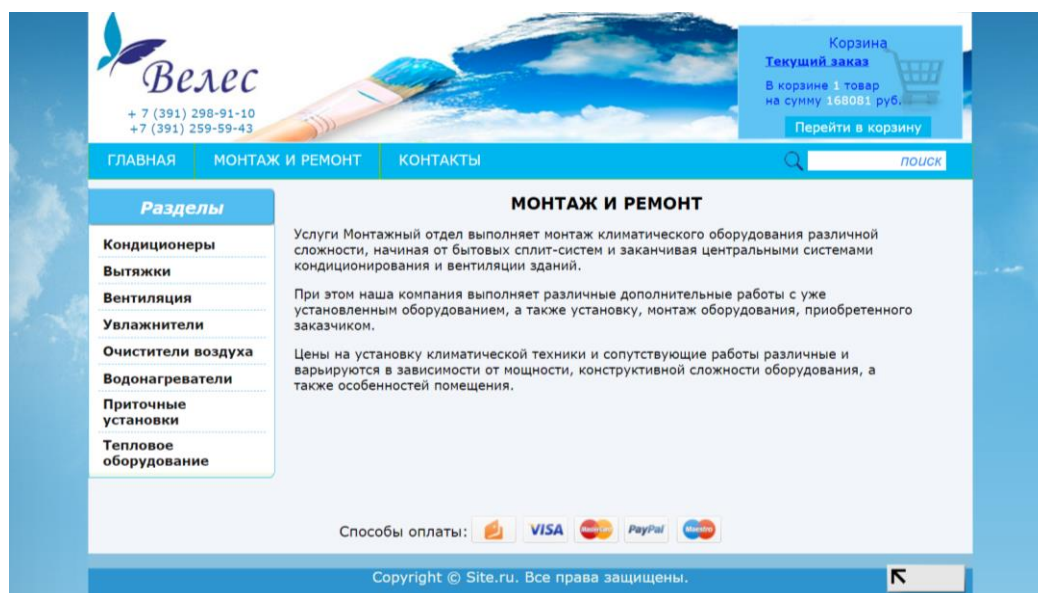




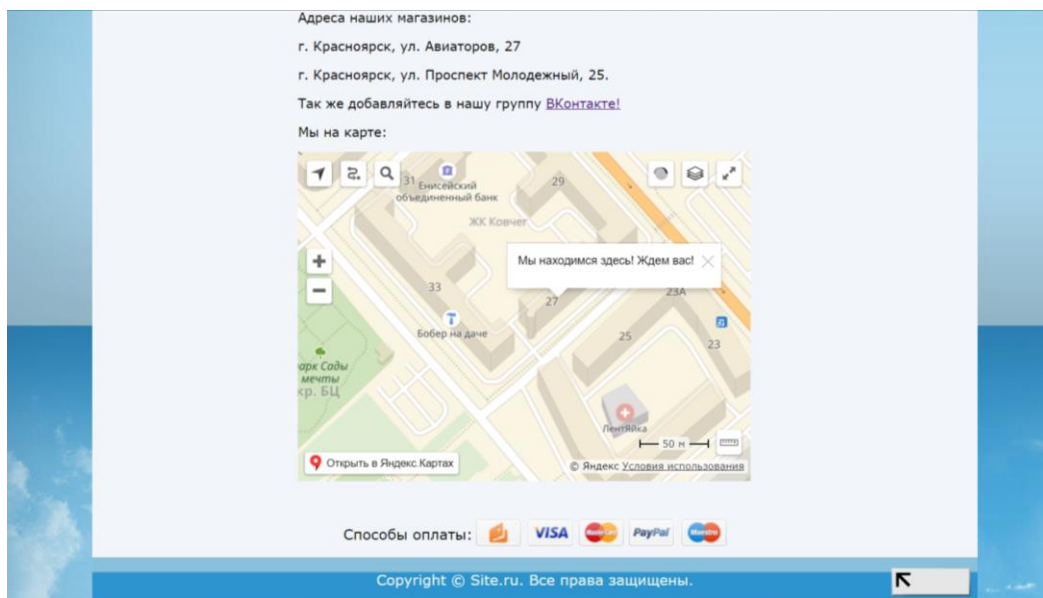
## Кросс-продажи сайта ООО «Велес Монтаж»



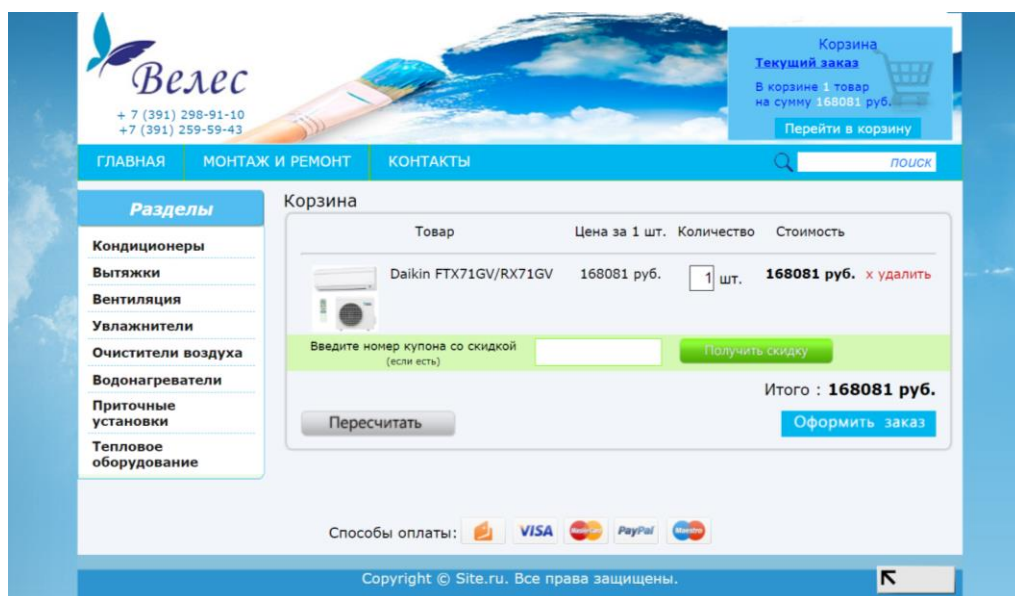
## Страница «Монтаж и Ремонт» сайта ООО «Велес Монтаж»



## Интерактивная Яндекс карта на странице «Контакты» сайта ООО «Велес Монтаж»



## Корзина сайта ООО «Велес Монтаж»



## Оформление заказа (с доставкой) ООО «Велес Монтаж»

The screenshot shows the 'Оформление заказа' (Order Form) page on the 'Велес Монтаж' website. The page has a blue header with navigation links: 'ГЛАВНАЯ', 'МОНТАЖ И РЕМОНТ', and 'КОНТАКТЫ'. A search bar is located in the top right corner. On the left side, there is a sidebar with a 'Разделы' (Sections) menu listing various services: 'Кондиционеры', 'Вытяжки', 'Вентиляция', 'Увлажнители', 'Очистители воздуха', 'Водонагреватели', 'Приточные установки', and 'Тепловое оборудование'. The main content area is titled 'Оформление заказа' and contains several input fields: 'ФИО:' with the value 'Якимова Мария', 'Телефон:' with '855577778', 'E-mail:' with 'myma@mail.ru', and 'Доставка:' with a dropdown menu showing 'Доставка(цена устанавлива' and a downward arrow. Below these fields is a large text area for 'Полный адрес (Город, улица, дом, квартира):' and another for 'Примечание к заказу:'. At the bottom right of the form is a blue button labeled 'Закончить оформление заказа'.

## Оформление заказа (самовывоз) ООО «Велес Монтаж»

This screenshot shows the same 'Оформление заказа' page as the previous one, but with the 'Доставка:' dropdown menu set to 'Самовывоз'. The rest of the form, including the contact information and address fields, remains the same. At the bottom of the page, there is a section for 'Способы оплаты:' (Payment Methods) with icons for 'VISA', 'MasterCard', 'PayPal', and 'WebMoney'. The footer contains the text 'Copyright © Site.ru. Все права защищены.' and a small logo.

## Вход в панель администратора

### Аккаунт администратора

#### Вход в Admin-панель

Логин:




Пароль:

Главная страница в панели администратора

Аккаунт администратора

Меню			
<a href="#">Главная</a>	<a href="#">Товары</a>	<a href="#">Заказы</a>	
<a href="#">Разделы</a>	<a href="#">Купоны</a>	<a href="#">Статистика</a>	
		<a href="#">Выход</a>	
Статистика за последние 7 дней			
Количество заказов	Счетов на сумму	Доход	Купленных товаров
2	117677 рублей	79696 рублей	4

Страница «Товары» в панели администратора

Аккаунт администратора						
Меню						
<a href="#">Главная</a>	<a href="#">Товары</a>	<a href="#">Заказы</a>				
<a href="#">Разделы</a>	<a href="#">Купоны</a>	<a href="#">Статистика</a>				
		<a href="#">Выход</a>				
Товары						
<a href="#">1</a> <a href="#">2</a> <a href="#">3</a> <a href="#">4</a> <a href="#">5</a>						
<a href="#">Добавить новый товар</a>						
ID	Раздел	Название	Цена	Информация	Дата добавления	Функции
43	Увлажнители	Neoclima NHL-NHL-220L (Черный) 	4387 рублей	Марка: Тип: Вес: Площадь обслуживания:	2018-06-03 20:22:54	<a href="#">Редактировать</a> <a href="#">Удалить</a>
42	Увлажнители	NEOCLIMA NHL-220L Красный 	2690 рублей	Марка: Тип: Вес: Площадь обслуживания:	2018-06-03 20:22:53	<a href="#">Редактировать</a> <a href="#">Удалить</a>
		BP 300-45-3,15 1,5кВт 		Марка: Тип:		<a href="#">Редактировать</a> <a href="#">Удалить</a>

Страница «Заказы» в панели администратора

Аккаунт администратора

Меню

[Главная](#)

[Товары](#)

[Заказы](#)

[Разделы](#)

[Купоны](#)

[Статистика](#)

[Выход](#)

Заказы

1

[Добавить новый заказ](#)

ID	Способ доставки	Заказ	Цена	ФИО	Телефон	E-mail	Адрес	Примечание	Дата заказа	Дата отправки	Дата оплаты	Функции
27	Самовывоз	Daikin FTX71GV/RX71GV (1 шт.)	168081	Якимова Мария	855577778	myta@mail.ru			2018-06-11 04:40:51	Не отправлено	Не оплачено	<a href="#">Редактировать</a> <a href="#">Удалить</a>
26	Самовывоз	Kentatsu KSGR35HFAN1/ KSRR35HFAN1 (1 шт.) Neoclima NS/NU-24TA5	79696	ринрат	8555	myta@mail.ru		qd	2018-06-09 12:34:00	2018-06-09 12:34:00	2018-06-11 01:25:14	<a href="#">Редактировать</a> <a href="#">Удалить</a>

Страница редактирования заказа на примере

Аккаунт администратора

Меню

[Главная](#)

[Товары](#)

[Заказы](#)

[Разделы](#)

[Купоны](#)

[Статистика](#)

[Выход](#)

Редактирование заказа

Заказ

Kentatsu KSGR35HFAN1 / KSRR35HFAN1

шт. 1

Удалить позицию

Neoclima NS/NU-24TA5

шт. 1

Удалить позицию

Добавить позицию

Способ доставки: 

Самовывоз

Цена: 79696

рублей

ФИО: ритрат

Телефон: 8555

E-mail: myma@mail.ru

Адрес:

Примечание:

qd

Отправлено: 

Отправлено

Оплачено: 

Оплачено

Отправить

Страница «Разделы» в панели администратора

Аккаунт администратора

Меню

[Главная](#)

[Товары](#)

[Заказы](#)

[Разделы](#)

[Купоны](#)

[Статистика](#)

[Выход](#)

Разделы

1

Добавить новый раздел

ID	Название	Функции
1	Кондиционеры	<a href="#">Редактировать</a> <a href="#">Удалить</a>
2	Вытяжки	<a href="#">Редактировать</a> <a href="#">Удалить</a>
3	Вентиляция	<a href="#">Редактировать</a> <a href="#">Удалить</a>
4	Увлажнители	<a href="#">Редактировать</a> <a href="#">Удалить</a>
5	Очистители воздуха	<a href="#">Редактировать</a> <a href="#">Удалить</a>
6	Водонагреватели	<a href="#">Редактировать</a> <a href="#">Удалить</a>
7	Приточные установки	<a href="#">Редактировать</a> <a href="#">Удалить</a>

101

## Страница «Купоны» в панели администратора

Аккаунт администратора

Меню

[Главная](#)  
[Разделы](#)

[Товары](#)  
[Купоны](#)

[Заказы](#)  
[Статистика](#)  
[Выход](#)

Купоны

1

[Добавить новый купон](#)

ID	Код	Размер скидки	Функции
1	ABC	10 %	<a href="#">Редактировать</a> <a href="#">Удалить</a>

## Страница «Статистика» в панели администратора

Аккаунт администратора

Меню

[Главная](#)  
[Разделы](#)

[Товары](#)  
[Купоны](#)

[Заказы](#)  
[Статистика](#)  
[Выход](#)

Статистика

От: 04.06.2018

До: 10.06.2018

Показать

Результат

Количество заказов	Счетов на сумму	Доход	Купленных товаров
2	117677 рублей	79696 рублей	4

## ПРИЛОЖЕНИЕ Б

### Листинг файла database\_class.php.

```
<?php
require_once "config_class.php";
class DataBase {
private static $db = null;
private $config;
private $mysqli;
public static function getDB() {
if (self::$db == null) self::$db = new DataBase();
return self::$db;
} private function __construct() {
$this->config = new Config();
$this->mysqli = new mysqli($this->config->db_host, $this->config->db_user, $this->config->db_password, $this->config->db_name);
$this->mysqli->query("SET NAMES 'utf8'");
}private function getQuery($query, $params) {
if ($params) {
for ($i = 0; $i < count($params); $i++) {
$pos = strpos($query, $this->config->sym_query);
$arg = "".$this->mysqli->real_escape_string($params[$i])."";
$query = substr_replace($query, $arg, $pos, strlen($this->config->sym_query));
}
}
return $query;
}
public function select($query, $params = false) {
$result_set = $this->mysqli->query($this->getQuery($query, $params));
if (!$result_set) return false;
return $this->resultSetToArray($result_set);
}
public function selectRow($query, $params = false) {
$result_set = $this->mysqli->query($this->getQuery($query, $params));
if ($result_set->num_rows != 1) return false;
return $result_set->fetch_assoc();
}
public function selectCell($query, $params = false) {
$result_set = $this->mysqli->query($this->getQuery($query, $params));
if (!$result_set) || ($result_set->num_rows != 1) return false;
else {
$arr = array_values($result_set->fetch_assoc());
return $arr[0];
}
}
public function query($query, $params = false) {
$success = $this->mysqli->query($this->getQuery($query, $params));
if ($success) {
if ($this->mysqli->insert_id === 0) return true;
else return $this->mysqli->insert_id;
}
else return false;
}
private function resultSetToArray($result_set) {
$array = array();
while (($row = $result_set->fetch_assoc()) != false) {
$array[] = $row;
}
return $array;
}
public function __destruct() {
if ($this->mysqli) $this->mysqli->close();
}
```

```

    }
}
?>

```

**Листинг файла format\_class.php.**

```

<?php
require_once "config_class.php";
class Format {
    private $config;

    public function __construct() {
        $this->config = new Config();
    }
    public function ts() {
        return time();
    }
    public function date($t) {
        return date("Y-m-d H:i:s", $t);
    }
    public function time($ts) {
        return date("d.m.Y", $ts);
    }
    public function xss($data) {
        if (is_array($data)) {
            $escaped = array();
            foreach ($data as $key => $value) {
                $escaped[$key] = $this->xss($value);
            }
            return $escaped;
        }
        return htmlspecialchars($data);
    }
    public function getTime($t, $start) {
        if ($t) {
            preg_match_all("/(\d{2}).(\d{2}).(\d{4})/i", $t, $matches);
            if ($start) return mktime(0, 0, 0, $matches[2][0], $matches[1][0], $matches[3][0]);
            else return mktime(23, 59, 59, $matches[2][0], $matches[1][0], $matches[3][0]);
        }
        else {
            if ($start) return mktime(0, 0, 0, date("n"), date("j"), date("Y")) - 7 * 24 * 3600;
            else return mktime(23, 59, 59, date("n"), date("j"), date("Y")) - 24 * 3600;
        }
    }
    public function hash($str) {
        return md5($str.$this->config->secret);
    }
}
?>

```

**Листинг файла url\_class.php.**

```

<?php
require_once "config_class.php";
class URL {
    protected $config;
    protected $amp;
    public function __construct($amp = true) {
        $this->config = new Config();
        $this->amp = $amp;
    }
    public function getView() {
        $view = $_SERVER["REQUEST_URI"];
        $view = substr($view, 1);
        if (($pos = strpos($view, "?")) !== false) {

```



```

        $view = substr($view, 0, $pos);
    }
    return $view;
}
public function setAMP($amp) {
    $this->amp = $amp;
}
public function getThisURL() {
    $uri = substr($_SERVER["REQUEST_URI"], 1);
    return $this->config->address.$uri;
}
protected function deleteGET($url, $param) {
    $res = $url;
    if (($p = strpos($res, "?")) !== false) {
        $paramstr = substr($res, $p + 1);
        $params = explode("&", $paramstr);
        $paramsarr = array();
        foreach ($params as $value) {
            $tmp = explode("=", $value);
            $paramsarr[$tmp[0]] = $tmp[1];
        }
        if (array_key_exists($param, $paramsarr)) {
            unset($paramsarr[$param]);
            $res = substr($res, 0, $p + 1);
            foreach ($paramsarr as $key => $value) {
                $str = $key;
                if ($value !== "") {
                    $str .= "=$value";
                }
                $res .= "$str&";
            }
            $res = substr($res, 0, -1);
        }
    }
    return $res;
}
public function index() {
    return $this->returnURL("");
}
public function addorder($id) {
    return $this->returnURL("addorder?id=$id");
}
public function notFound() {
    return $this->returnURL("notfound");
}
public function cart() {
    return $this->returnURL("cart");
}
public function order() {
    return $this->returnURL("order");
}
public function message() {
    return $this->returnURL("message");
}
public function delivery() {
    return $this->returnURL("delivery");
}
public function contacts() {
    return $this->returnURL("contacts");
}
public function search() {
    return $this->returnURL("search");
}

```

```

    }
    public function section($id) {
        return $this->returnURL("section?id=$id");
    }
    public function product($id) {
        return $this->returnURL("product?id=$id");
    }
    public function addCart($id) {
        return $this->returnURL("functions.php?func=add_cart&id=$id");
    }
    public function deleteCart($id) {
        return $this->returnURL("functions.php?func=delete_cart&id=$id");
    }
    public function sortTitleUp() {
        return $this->sortOnField("title", 1);
    }
    public function sortTitleDown() {
        return $this->sortOnField("title", 0);
    }
    public function sortPriceUp() {
        return $this->sortOnField("price", 1);
    }
    public function sortPriceDown() {
        return $this->sortOnField("price", 0);
    }
    public function action() {
        return $this->returnURL("functions.php");
    }
    public function adminEditProduct($id) {
        return $this->returnURL("?view=products&func=edit&id=$id", $this->config->address_admin);
    }
    public function adminDeleteProduct($id) {
        return $this->returnURL("functions.php?func=delete_product&id=$id", $this->config->address_admin);
    }
    public function adminEditSection($id) {
        return $this->returnURL("?view=sections&func=edit&id=$id", $this->config->address_admin);
    }
    public function adminDeleteSection($id) {
        return $this->returnURL("functions.php?func=delete_section&id=$id", $this->config->address_admin);
    }
    public function adminEditOrder($id) {
        return $this->returnURL("?view=orders&func=edit&id=$id", $this->config->address_admin);
    }
    public function adminDeleteOrder($id) {
        return $this->returnURL("functions.php?func=delete_order&id=$id", $this->config->address_admin);
    }
    public function adminEditDiscount($id) {
        return $this->returnURL("?view=discounts&func=edit&id=$id", $this->config->address_admin);
    }
    public function adminDeleteDiscount($id) {
        return $this->returnURL("functions.php?func=delete_discount&id=$id", $this->config->address_admin);
    }
    protected function sortOnField($field, $up) {
        $this_url = $this->getThisURL();
        $this_url = $this->deleteGET($this_url, "up");
        $this_url = $this->deleteGET($this_url, "sort");
        if (strpos($this_url, "?") === false) $url = $this->url."?sort=$field&up=$up";
        else $url = $this_url."&sort=$field&up=$up";
        return $this->returnURL($url);
    }
    protected function returnURL($url, $index = false) {
        if (!$index) $index = $this->config->address;
    }

```

```

        if ($url == "") return $index;
        if (strpos($url, $index) !== 0) $url = $index.$url;
        if ($this->amp) $url = str_replace("&", "& ", $url);
        return $url;
    }
    public function fileExists($file) {
        $arr = explode(PATH_SEPARATOR, get_include_path());
        foreach ($arr as $val) {
            if (file_exists($val."/".$file)) return true;
        }
        return false;
    }
}
?>

```

#### Листинг файла **check\_class.php**

```

<?php
require_once "config_class.php";
class Check {
    private $config;
    public function __construct($amp = true) {
        $this->config = new Config();
    }
    public function id($id, $zero = false) {
        if (!$this->intNumber($id)) return false;
        if (!$zero && ($id === 0)) return false;
        return $id >= 0;
    }
    public function value($value) {
        if (!$this->doubleNumber($value)) return false;
        return ($value > 0 && $value <= 1);
    }
    public function ids($ids) {
        $reg = "/^\\d+(,\\d+)*\\d?$/i";
        return preg_match($reg, $ids);
    }
    public function amount($amount) {
        if (!$this->doubleNumber($amount)) return false;
        return $amount >= 0;
    }
    public function name($name) {
        if ($this->isContainQuotes($name)) return false;
        return $this->isString($name, 1, $this->config->max_name);
    }
    public function title($title) {
        return $this->isString($title, 1, $this->config->max_title);
    }
    public function email($email) {
        if ($this->isContainQuotes($email)) return false;
        $reg = "/^[a-z0-9][a-z0-9\\._-]*[a-z0-9_]*@([a-z0-9]+([a-z0-9-]*[a-z0-9]+)*\\.)+[a-z]+$/i";
        return preg_match($reg, $email);
    }
    public function text($text, $empty = false) {
        if (($empty) && ($text == "")) return true;
        return $this->isString($text, 1, $this->config->max_text);
    }
    public function ts($ts) {
        return $this->noNegativeInteger($ts);
    }
    public function year($year) {
        if (!$this->intNumber($year)) return false;
        return $year > 0;
    }
}

```

```

    }
    }
    public function oneOrZero($number) {
        if (!$this->intNumber($number)) return false;
        return (($number == 0) || ($number == 1));
    }
    public function count($count) {
        return $this->noNegativeInteger($count);
    }
    public function offset($offset) {
        return $this->intNumber($offset);
    }
    private function noNegativeInteger($number) {
        if (!$this->intNumber($number)) return false;
        return ($number >= 0);
    }
    private function intNumber($number) {
        if (!is_int($number) && (!is_string($number))) return false;
        return preg_match("/^-?([1-9][0-9]*)|(0)$/", $number);
    }
    private function isContainQuotes($string) {
        $array = array("\"", "'", "^", "&quot;", "&apos;");
        foreach ($array as $value) {
            if (strpos($string, $value) !== false) return true;
        }
        return false;
    }
    private function isString($string, $min_length, $max_length) {
        if (!is_string($string)) return false;
        if (strlen($string) < $min_length) return false;
        if (strlen($string) > $max_length) return false;
        return true;
    }
    private function doubleNumber($number) {
        return is_numeric($number);
    }
}
?>

```

#### Листинг файла **global\_class.php**

```

<?php
require_once "database_class.php";
require_once "config_class.php";
require_once "check_class.php";
require_once "url_class.php";
require_once "systemmessage_class.php";
abstract class GlobalClass {
    protected $db;
    protected $table_name;
    protected $format;
    protected $config;
    protected $check;
    protected $url;
    public function __construct($table_name) {
        $this->db = DataBase::getDB();
        $this->format = new Format();
        $this->config = new Config();
        $this->check = new Check();
        $this->url = new URL();
        $this->table_name = $this->config->db_prefix.$table_name;
    }
    public function add($data) {

```

```

        if (!$this->check($data)) return false;
        $query = "INSERT INTO `". $this->table_name. "` (" ;
        foreach ($data as $field => $value) $query .= "`$field`,";
        $query = substr($query, 0, -1);
        $query .= ") VALUES (" ;
        foreach ($data as $value) $query .= $this->config->sym_query. ",";
        $query = substr($query, 0, -1);
        $query .= ")";
        return $this->db->query($query, array_values($data));
    }

    public function edit($id, $data) {
        if (!$this->check($data)) return false;
        $query = "UPDATE `". $this->table_name. "` SET ";
        foreach ($data as $field => $value) $query .= "`$field` = ".$this->config->sym_query. ",";
        $query = substr($query, 0, -1);
        $query .= "WHERE `id` = ".$this->config->sym_query;
        $data["id"] = $id;
        return $this->db->query($query, array_values($data));
    }

    public function delete($id) {
        $query = "DELETE FROM `". $this->table_name. "` WHERE `id` = ".$this->config->sym_query;
        return $this->db->query($query, array($id));
    }

    private function check($data) {
        $result = $this->checkData($data);
        if ($result === true) return true;
        $sm = new SystemMessage();
        return $sm->message($result);
    }

    protected function checkData($data) {
        return false;
    }

    public function existsID($id) {
        if (!$this->check->id($id)) return false;
        return $this->isExistsFV("id", $id);
    }

    protected function isExistsFV($field, $value) {
        $result = $this->getAllOnField($field, $value);
        return count($result) != 0;
    }

    protected function getAll($order = false, $up = true, $count = false, $offset = false) {
        $sol = $this->getOL($order, $up, $count, $offset);
        $query = "SELECT * FROM `". $this->table_name. "` $sol";
        return $this->db->select($query);
    }

    public function get($id) {
        if (!$this->check->id($id)) return false;
        return $this->getOnField("id", $id);
    }

    protected function getField($field_in, $value_in, $field_out) {
        $query = "SELECT `$field_out` FROM `". $this->table_name. "` WHERE `$field_in` = ".$this->
>config->sym_query;
        return $this->db->selectCell($query, array($value_in));
    }

    protected function getFieldOnID($id, $field) {
        return $this->getField("id", $id, $field);
    }

    protected function getOnField($field, $value) {
        $query = "SELECT * FROM `". $this->table_name. "` WHERE `$field` = ".$this->config->
>sym_query;
        return $this->db->selectRow($query, array($value));
    }

```

```

protected function getAllOnField($field, $value, $order = false, $sup = true, $count = false, $offset
= false) {
    $ol = $this->getOL($order, $sup, $count, $offset);
    $query = "SELECT * FROM `".$this->table_name."` WHERE `".$field.` = ".$this->config-
>sym_query." $ol";
    return $this->db->select($query, array($value));
}
protected function getOL($order, $sup, $count, $offset) {
    if ($order) {
        $order = "ORDER BY `".$order.`";
        if (!$sup) $order .= " DESC";
    }
    $limit = $this->getL($count, $offset);
    return "$order $limit";
}
protected function setField($field_in, $value_in, $field, $value) {
    $query = "UPDATE `".$this->table_name."` SET `".$field.` = ".$this->config->sym_query." WHERE
`".$field_in.` = ".$this->config->sym_query;
    return $this->db->query($query, array($value, $value_in));
}
protected function setFieldOnID($id, $field, $value) {
    return $this->setField("id", $id, $field, $value);
}
protected function transform($element) {
    if (!$element) return false;
    if (isset($element[0])) {
        for ($i = 0; $i < count($element); $i++)
            $element[$i] = $this->transformElement($element[$i]);
        return $element;
    }
    else return $this->transformElement($element);
}
protected function getL($count, $offset) {
    $limit = "";
    if ($count) {
        if (!$this->check->count($count)) return false;
        if ($offset) {
            if (!$this->check->offset($offset)) return false;
            $limit = "LIMIT $offset, $count";
        }
        else $limit = "LIMIT $count";
    }
    return $limit;
}
public function getCount() {
    $query = "SELECT COUNT(id) FROM `".$this->table_name."`;
    return $this->db->selectCell($query);
}
public function search($q, $fields, $order = false, $sup = false) {
    if (count($fields) == 0) return false;
    $q = trim($q);
    if ($q === "") return false;
    $q = preg_replace("/\s+/", " ", $q);
    $q = mb_strtolower($q);
    $array_words = explode(" ", $q);
    $logic = " AND ";
    $params = array();
    foreach ($array_words as $key => $value) {
        if (isset($array_words[$key - 1])) $where .= $logic;
        for ($i = 0; $i < count($fields); $i++) {
            $where .= "`".$fields[$i]."` LIKE ".$this->config->sym_query;
            $params[] = "%$value%";
        }
    }
}

```

```

        if (($i + 1) != count($fields)) $where .= " OR ";
    }
}
$ol = $this->getOL($order, $sup, 0, 0);
$query = "SELECT * FROM `".$this->table_name."` WHERE $where $ol";
return $this->db->select($query, $params);
}
public function getTableName() {
    return $this->table_name;
}
}
?>

```

#### Листинг файла **section\_class.php**

```

<?php
require_once "global_class.php";
class Section extends GlobalClass {
    public function __construct() {
        parent::__construct("sections");
    }
    public function getAllData() {
        return $this->transform($this->getAll("id"));
    }
    public function getTableData($count, $offset) {
        return $this->transform($this->getAll("id", true, $count, $offset));
    }
    public function get($id) {
        return $this->transform(parent::get($id));
    }
    protected function transformElement($section) {
        $section["link"] = $this->url->section($section["id"]);
        $section["link_admin_edit"] = $this->url->adminEditSection($section["id"]);
        $section["link_admin_delete"] = $this->url->adminDeleteSection($section["id"]);
        return $section;
    }
    protected function checkData($data) {
        if (!$this->check->title($data["title"])) return "ERROR_TITLE";
        return true;
    }
}
?>

```

#### Листинг файла **discount\_class.php**

```

<?php
require_once "global_class.php";
class Discount extends GlobalClass {
    public function __construct() {
        parent::__construct("discounts");
    }
    public function getOnCode($code) {
        return $this->getOnField("code", $code);
    }
    public function getTableData($count, $offset) {
        return $this->transform($this->getAll("id", true, $count, $offset));
    }
    protected function transformElement($discount) {
        $discount["link_admin_edit"] = $this->url->adminEditDiscount($discount["id"]);
        $discount["link_admin_delete"] = $this->url->adminDeleteDiscount($discount["id"]);
        return $discount;
    }
    public function getValueOnCode($code) {
        return $this->getField("code", $code, "value");
    }
}

```

```

    }
    protected function checkData($data) {
        if (!$this->check->title($data["code"])) return "ERROR_CODE";
        if (!$this->check->value($data["value"])) return "ERROR_VALUE";
        return true;
    }
}
?>

```

#### Листинг файла `order_class.php`

```

<?php
require_once "global_class.php"
class Order extends GlobalClass {
    public function __construct() {
        parent::__construct("orders");
    }
    public function getAllInInterval($start, $end) {
        $query = "SELECT * FROM `".$this->table_name."` WHERE `date_order` > ".$this->config-
>sym_query." AND `date_order` < ".$this->config->sym_query;
        return $this->db->select($query, array($start, $end));
    }
    public function getProductIDs($id) {
        return $this->getFieldOnID($id, "product_ids");
    }
    public function getTableData($count, $offset) {
        return $this->transform($this->getAll("date_order", false, $count, $offset));
    }
    public function get($id) {
        return $this->transform(parent::get($id));
    }
    public function getPrice($id) {
        return $this->getFieldOnID($id, "price");
    }
    public function getDateOrder($id) {
        return $this->getFieldOnID($id, "date_order");
    }
    public function getDateSend($id) {
        return $this->getFieldOnID($id, "date_send");
    }
    public function getDatePay($id) {
        return $this->getFieldOnID($id, "date_pay");
    }
    public function setDatePay($id, $date_pay) {
        if (!$this->check->ts($date_pay)) return "UNKNOWN_ERROR";
        return $this->setFieldOnID($id, "date_pay", $date_pay);
    }
    protected function transformElement($order) {
        $order["link_admin_edit"] = $this->url->adminEditOrder($order["id"]);
        $order["link_admin_delete"] = $this->url->adminDeleteOrder($order["id"]);
        $order["date_order"] = $this->format->date($order["date_order"]);
        if ($order["date_send"] != 0) $order["date_send"] = $this->format->date($order["date_send"]);
        if ($order["date_pay"] != 0) $order["date_pay"] = $this->format->date($order["date_pay"]);
        return $order;
    }
    protected function checkData($data) {
        if (!$this->check->oneOrZero($data["delivery"])) return "ERROR_DELIVERY";
        if (!$this->check->ids($data["product_ids"])) return "UNKNOWN_ERROR";
        if (!$this->check->amount($data["price"])) return "ERROR_PRICE";
        if (!$this->check->name($data["name"])) return "ERROR_NAME";
        if (!$this->check->title($data["phone"])) return "ERROR_PHONE";
        if (!$this->check->email($data["email"])) return "ERROR_EMAIL";
        if ($data["delivery"] == 1) $empty = true;
    }
}

```



```

else $empty = false;
if (!$this->check->text($data["address"], $empty)) return "ERROR_ADDRESS";
if (!$this->check->text($data["notice"], true)) return "ERROR_NOTICE";
if (!$this->check->ts($data["date_order"])) return "UNKNOWN_ERROR";
if (!$this->check->ts($data["date_send"])) return "UNKNOWN_ERROR";
if (!$this->check->ts($data["date_pay"])) return "UNKNOWN_ERROR";
return true;
}
}
?>

```

#### Листинг файла main.tpl

```

<!DOCTYPE html>
<html >
<head>
    <title><?=$this->title?></title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta name="description" content="<?=$this->meta_desc?>" />
    <meta name="keywords" content="<?=$this->meta_key?>" />
    <link rel="stylesheet" href="styles/main.css" type="text/css" />

    <script type="text/javascript" src="js/functions.js"></script>
    <link href="favicon.ico" rel="shortcut icon" type="image/x-icon" />
</head>
<body>
    <div id="container">
        <div id="header">
            
        </div>
        <p class="red">8-800-123-45-67</p>
        <p class="blue">Время работы с 09:00 до 21:00<br />без перерыва и выходных</p>
        </div>
        <div class="cart">
            <p class="cart_title">Корзина</p>
            <p class="blue">Текущий заказ</p>
            <p>В корзине <span><?=$this->cart_count?></span> <?=$this->
>cart_word?><br />на сумму <span><?=$this->cart_summa?></span> руб.</p>
            <a href="<?=$this->cart_link?>">Перейти в корзину</a>
        </div>
        </div>
        <div id="topmenu">
            <ul>
                <li>
                    <a href="<?=$this->index?>">ГЛАВНАЯ</a>
                </li>
                <li>
                    
                </li>
                <li>
                    <a href="<?=$this->link_delivery?>">МОНТАЖ И РЕМОУНТ</a>
                </li>
                <li>
                    
                </li>
                <li>
                    <a href="<?=$this->link_contacts?>">КОНТАКТЫ</a>
                </li>
            </ul>
        </div>
        <div id="search">
            <form name="search" action="<?=$this->link_search?>" method="get">
            <table>
            <tr>

```

```

        <td class="input_left"></td>
        <td>
            <input type="text" name="q" value="поиск" onfocus="if(this.value == 'поиск')
this.value=''" onblur="if(this.value == '') this.value='поиск'" />
        </td>
        <td class="input_right"></td>
    </tr>
</table>
</form>
</div>
</div>
<div id="content">
<div id="left">
<div id="menu">
<div class="header">
<h3>Разделы</h3>
</div>
<div id="items">
<?php for ($i = 0; $i < count($this->items); $i++) { ?>
<p <?php if ($i + 1 == count($this->items)) { ?>class="last"<?php } ?>>
<a href="<?=$this->items[$i]["link"] ?>"><?=$this->items[$i]["title"] ?></a>
</p>
<?php } ?>
</div>
<div class="bottom"></div>
    </div>
</div>
<div id="right">
    <?php include "content_". $this->content. ".tpl"; ?>
</div>
<div class="clear"></div>
<div id="footer">
<div id="pm">
<table>
<tr>
<td>Способы оплаты:</td>
<td>

</td>
</tr>
</table>
</div>
<div id="copy">
<p>Copyright &copy; Site.ru. Все права защищены.</p>
<p class="counter">
<!--LiveInternet counter--><script type="text/javascript"><!--
document.write("<a href='http://www.liveinternet.ru/click' "+
"target=_blank"></a>") //--></script><!--/LiveInternet-->
</p>
</div>
</div>
</div>
</div>
</body>
</html>

```

### Листинг файла content\_order.tpl

```
<div id="order">
    <h2>Оформление заказа</h2>
    <?php include "message.tpl"; ?>
    <form name="order" action="<?=$this->action?>" method="post">
        <table>
            <tr>
                <td class="w120">ФИО:</td>
                <td>
                    <input type="text" name="name" value="<?=$this->name?>" />
                </td>
            </tr>
            <tr>
                <td>Телефон:</td>
                <td>
                    <input type="text" name="phone" value="<?=$this->phone?>" />
                </td>
            </tr>
            <tr>
                <td>E-mail:</td>
                <td>
                    <input type="text" name="email" value="<?=$this->email?>" />
                </td>
            </tr>
            <tr>
                <td>Доставка:</td>
                <td>
                    <select name="delivery" onchange="changeDelivery(this)">
                        <option value="">выберите, пожалуйста...</option>
                        <option value="0" <?php if ($this->delivery == "0") { ?>selected="selected"<?php
}??>Доставка(цена устанавливается индивидуально)</option>
                        <option value="1" <?php if ($this->delivery == "1") { ?>selected="selected"<?php
}??>Самовывоз</option>
                    </select>
                </td>
            </tr>
        </table>
        <table>
            <tr id="address">
                <td>
                    <p>Полный адрес (Город, улица, дом, квартира):</p>
                    <textarea name="address" cols="80" rows="6"><?=$this->address?></textarea>
                </td>
            </tr>
            <tr>
                <td>
                    <p>Примечание к заказу:</p>
                    <textarea name="notice" cols="80" rows="6"><?=$this->notice?></textarea>
                </td>
            </tr>
            <tr>
                <td class="button">
                    <input type="hidden" name="func" value="order" />
                    <input type="image" src="images/order_end.png" alt="Закончить оформление заказа"
onmouseover="this.src='images/order_end_active.png'" onmouseout="this.src='images/order_end.png'" />
                </td>
            </tr>
        </table>
    </form>
</div>
```

## Листинг файла content\_product.tpl

```
<table id="hornav">
    <tr>
        <td>
            <a href="<?=$this->index?>">Главная</a>
        </td>
        <td>
            
        </td>
        <td>
            <a href="<?=$this->link_section?>"><?=$this->product["section"]?></a>
        </td>
        <td>
            
        </td>
        <td><?=$this->product["title"]?></td>
    </tr>
</table>
<table id="product">
    <tr>
        <td class="product_img">
            " alt="<?=$this->product["title"]?>" />
        </td>
        <td class="product_desc">
            <p>Название: <span class="title"><?=$this->product["title"]?></span></p>
            <p>Марка: <span><?=$this->product["brand"]?></span></p>
            <p>Раздел: <span><?=$this->product["section"]?></span></p>
            <p>Тип: <span><?=$this->product["type"]?></span></p>
            <p>Вес: <span><?=$this->product["wt"]?></span></p>
            <p>Площадь обслуживания: <span><?=$this->product["area"]?></span></p>
            <p>Производительность: <span><?=$this->product["volt"]?></span></p>
        </td>
    </tr>
    <tr>
        <td>
            <p class="price"><?=$this->product["price"]?> руб.</p>
        </td>
        <td>
            <p>
                <a class="link_cart" href="<?=$this->product["link_cart"]?>"></a>
            </p>
        </td>
    </tr>
</table>
</td>
</tr>
<tr>
    <td colspan="2">
        <p class="desc_title">Описание:</p>
        <p class="desc"><?=$this->product["description"]?></p>
    </td>
</tr>
</table>
<div id="others">
    <h3>С этим товаром также заказывают:</h3>
    <table class="products">
        <tr>
            <?php for ($i = 0; $i < count($this->products); $i++) { ?>
            <td>
                <div class="intro_product">
                    <p class="img">
                        " alt="<?=$this->products[$i]["title"]?>" />
                    </p>
                </div>
            </td>
        </tr>
    </table>
</div>
```

```

</p>
<p class="title">
<a href="<?=$this->products[$i]["link"]?>"><?=$this->products[$i]["title"]?></a>
</p>
<p class="price"><?=$this->products[$i]["price"]?> руб.</p>
<p>
<a class="link_cart" href="<?=$this->products[$i]["link_cart"]?>"></a>
</p>
</div>
</td>
<?php } ?>
</tr>
</table>
</div>

```

#### Листинг файла content\_cart.tpl

```

<div id="cart">
<h2>Корзина</h2>
<form name="cart" action="<?=$this->action?>" method="post">
<table>
<tr>
<td colspan="8" id="cart_top"></td>
</tr>
<tr>
<td class="cart_left"></td>
<td colspan="2">Товар</td>
<td>Цена за 1 шт.</td>
<td>Количество</td>
<td>Стоимость</td>
<td></td>
<td class="cart_right"></td>
</tr>
<tr>
<td class="cart_left"></td>
<td colspan="6">
<hr />
<td class="cart_right"></td>
</tr>
<?php for ($i = 0; $i < count($this->cart); $i++) { ?>
<tr class="cart_row">
<td class="cart_left"></td>
<td class="img">
" alt="<?=$this->cart[$i]["title"]?>" />
</td>
<td class="title"><?=$this->cart[$i]["title"]?></td>
<td><?=$this->cart[$i]["price"]?> руб.</td>
<td>
<table class="count">
<tr>
<td>
<input type="text" name="count_<?=$this->cart[$i]["id"]?>" value="<?=$this->
>cart[$i]["count"]?>" />
<td>шт.</td>
</tr>
</table>
</td>
<td class="bold"><?=$this->cart[$i]["summa"]?> руб.</td>
<td>
<a href="<?=$this->cart[$i]["link_delete"]?>" class="link_delete">x
удалить</a>

```

```

</td>
<td class="cart_right"></td>
</tr>
<?php if ($i + 1 != count($this->cart)) { ?>
<tr>
<td class="cart_left"></td>
<td colspan="6" class="cart_border">
<hr />
</td>
<td class="cart_right"></td>
</tr>
<?php } ?>
<?php } ?>
<tr id="discount">
<td class="cart_left"></td>
<td colspan="6">
<form name="discount" action="#" method="post">
<table>
<tr>
<td>Введите номер купона со скидкой<br /><span>(если есть)</span></td>
<td>
<input type="text" name="discount" value="<?=$this->discount?" />
</td>
<td>
<input type="image" src="images/cart_discount.png" alt="Получить скидку"
onmouseover="this.src='images/cart_discount_active.png'" onmouseout="this.src='images/cart_discount.png'" />
</td>
</tr>
</table>
</form>
</td>
<td class="cart_right"></td>
</tr>
<tr id="summa">
<td class="cart_left"></td>
<td colspan="6">
<p>Итого <?php if ($this->discount) { ?>(с учётом скидки)<?php }?>:
<span><?=$this->summa?> руб.</span></p>
</td>
<td class="cart_right"></td>
</tr>
<tr>
<td class="cart_left"></td>
<td colspan="2">
<div class="left">
<input type="image" src="images/cart_recalc.png" alt="Пересчитать"
onmouseover="this.src='images/cart_recalc_active.png'" onmouseout="this.src='images/cart_recalc.png'" />
</div>
</td>
<td colspan="4">
<div class="right">
<input type="hidden" name="func" value="cart" />
<a href="<?=$this->link_order?">

</a>
</div>
</td>
<td class="cart_right"></td>
</tr>
<tr>
<td colspan="8" id="cart_bottom"></td>

```

```

        </tr>
    </table>
</form>
</div>

```

#### Листинг файла content\_notfound.tpl

```

<div id="message">
    <h2>Страница не существует</h2>
    <p>Возможно, Вы ошиблись при вводе адреса.<br />Вернуться на <a href="<?=$this-
>index?>">главную</a>.</p>
</div>

```

#### Листинг файла content\_search.tpl

```

<div id="search_result">
<?php if ($this->q == "") { ?>
    <h2>Вы задали пустой поисковый запрос!</h2>
<?php } else { ?>
    <h2>Результаты поиска: <?=$this->q?></h2>
    <?php if (!$this->products) { ?>
        <p>Ничего не найдено</p>
    <?php } else { ?>
        <?php include "table_products_top.tpl"; ?>
        <?php include "table_products.tpl"; ?>
    <?php } ?>
<?php } ?>
</div>

```

#### Листинг файла table\_product\_top.tpl

```

<table>
    <tr>
        <td rowspan="2">
            <div class="header">
                <h3><?=$this->table_products_title?></h3>
            </div>
        </td>
        <td class="section_bg"></td>
        <td class="section_right"></td>
    </tr>
    <tr>
        <td colspan="2">
            <table class="sort">
                <tr>
                    <td>Сортировать по:</td>
                    <td>цене (<a href="<?=$this->link_price_up?>">возр.</a> | <a href="<?=$this-
>link_price_down?>">убыв.</a>)
                    <td>названию (<a href="<?=$this->link_title_up?>">возр.</a> | <a href="<?=$this-
>link_title_down?>">убыв.</a>)
                </tr>
            </table>
        </td>
    </tr>
</table>

```

#### Листинг файла template\_class.tpl

```

<?php
class Template {
    private $dir_tmpl;
    private $data = array();
}

```

```

        public function __construct($dir_tmpl) {
            $this->dir_tmpl = $dir_tmpl;
        }
        public function set($name, $value) {
            $this->data[$name] = $value;
        }
        public function delete($name) {
            unset($this->data[$name]);
        }
        public function __get($name) {
            if (isset($this->data[$name])) return $this->data[$name];
            return "";
        }
        public function display($template) {
            $template = $this->dir_tmpl.$template.".tpl";
            ob_start();
            include ($template);
            echo ob_get_clean();
        }
    }
}
?>

```

#### Листинг файла abstractmodules\_class.php

```

<?php
require_once "config_class.php";
require_once "url_class.php";
require_once "format_class.php";
require_once "template_class.php";
require_once "section_class.php";
require_once "product_class.php";
require_once "discount_class.php";
require_once "order_class.php";
require_once "message_class.php";
abstract class AbstractModules {
    protected $config;
    protected $data;
    protected $url;
    protected $format;
    protected $section;
    protected $product;
    protected $discount;
    protected $order;
    protected $message;
    public function __construct() {
        session_start();
        $this->config = new Config();
        $this->url = new URL();
        $this->format = new Format();
        $this->data = $this->format->xss($_REQUEST);
        $this->template = new Template($this->getDirTpl());
        $this->section = new Section();
        $this->product = new Product();
        $this->discount = new Discount();
        $this->order = new Order();
        $this->message = new Message();
    }
    abstract protected function getContent();
    protected function notFound() {
        $this->redirect($this->url->notFound());
    }
    protected function message() {

```



```

        if (!$_SESSION["message"]) return "";
        $text = $this->message->get($_SESSION["message"]);
        unset($_SESSION["message"]);
        return $text;
    }
    protected function redirect($link) {
        header("Location: $link");
        exit;
    }
    abstract protected function getDirTpl();
    protected function getCountInArray($v, $array) {
        $count = 0;
        for ($i = 0; $i < count($array); $i++) {
            if ($array[$i] == $v) $count++;
        }
        return $count;
    }
}
?>

```

#### Листинг файла functions.php

```

<?php
    require_once "start.php";
    require_once "manage_class.php";
    require_once "url_class.php";
    $manage = new Manage();
    $url = new URL();
    $func = $_REQUEST["func"];
    if ($func == "add_cart") {
        $manage->addCart();
    }
    elseif ($func == "delete_cart") {
        $manage->deleteCart();
    }
    elseif ($func == "cart") {
        $manage->updateCart();
    }
    elseif ($func == "order") {
        $success = $manage->addOrder();
    }
    elseif ($func == "success_pay") {
        $success = $manage->successPay();
    }
    elseif ($func == "fail_pay") {
        $success = $manage->failPay();
    }
    elseif ($func == "status_pay") {
        $success = $manage->statusPay();
    }
    else exit;
    if ($success) {
        $link = $url->message();
    }
    else {
        $link = ($_SERVER["HTTP_REFERER"] != "")? $_SERVER["HTTP_REFERER"]:
$url->index();
    }
    header("Location: $link");
    exit;
?>

```

## Листинг файла manage\_class.php

```
<?php
require_once "config_class.php";
require_once "format_class.php";
require_once "product_class.php";
require_once "order_class.php";
require_once "discount_class.php";
require_once "systemmessage_class.php";
require_once "mail_class.php";
require_once "url_class.php"
class Manage {
    protected $config;
    protected $format;
    protected $product;
    protected $order;
    protected $discount;
    protected $url;
    public function __construct() {
        session_start();
        $this->config = new Config();
        $this->format = new Format();
        $this->product = new Product();
        $this->order = new Order();
        $this->discount = new Discount();
        $this->sm = new SystemMessage();
        $this->mail = new Mail();
        $this->url = new URL();
        $this->data = $this->format->xss($_REQUEST);
        $this->saveData();
    }
    private function saveData() {
        foreach ($this->data as $key => $value) $_SESSION[$key] = $value;
    }
    public function addCart($id = false) {
        if (!$id) $id = $this->data["id"];
        if (!$this->product->existsID($id)) return false;
        if ($_SESSION["cart"]) $_SESSION["cart"] .= ",$id";
        else $_SESSION["cart"] = $id;
    }
    public function deleteCart() {
        $id = $this->data["id"];
        $ids = explode(",", $_SESSION["cart"]);
        $_SESSION["cart"] = "";
        for ($i = 0; $i < count($ids); $i++) {
            if ($ids[$i] != $id) $this->addCart($ids[$i]);
        }
    }
    public function updateCart() {
        $_SESSION["cart"] = "";
        foreach ($this->data as $k => $v) {
            if (strpos($k, "count_") !== false) {
                $id = substr($k, strlen("count_"));
                for ($i = 0; $i < $v; $i++) $this->addCart($id);
            }
        }
        $_SESSION["discount"] = $this->data["discount"];
    }
    public function addOrder() {
        $temp_data = array();
        $temp_data["delivery"] = $this->data["delivery"];
        $temp_data["product_ids"] = $_SESSION["cart"];
    }
}
```

```

$temp_data["price"] = $this->getPrice();
$temp_data["name"] = $this->data["name"];
$temp_data["phone"] = $this->data["phone"];
$temp_data["email"] = $this->data["email"];
$temp_data["address"] = $this->data["address"];
$temp_data["notice"] = $this->data["notice"];
$temp_data["date_order"] = $this->format->ts();
$temp_data["date_send"] = 0;
$temp_data["date_pay"] = 0;
$id = $this->order->add($temp_data);
if ($id) {
    $send_data = array();
    $send_data["products"] = $this->getProducts();
    $send_data["name"] = $temp_data["name"];
    $send_data["phone"] = $temp_data["phone"];
    $send_data["email"] = $temp_data["email"];
    $send_data["address"] = $temp_data["address"];
    $send_data["notice"] = $temp_data["notice"];
    $send_data["price"] = $temp_data["price"];
    $to = $temp_data["email"];
    $this->mail->send($temp_data["email"], $send_data, "ORDER");
    header("Location: ".$this->url->addOrder($id));
    exit;
}
return false;
}
public function successPay() {
    return $this->sm->pageMessage("SUCCESS_PAY", true);
}
public function failPay() {
    return $this->sm->pageMessage("FAIL_PAY", true);
}
public function statusPay() {
    if ($this->data["ik_payment_state"] == "success") {
        $secret_key = "y2PlvVXsD7W13tWA";
        $sign = $this->data["ik_shop_id"].':'.
            $this->data["ik_payment_amount"].':'.
            $this->data["ik_payment_id"].':'.
            $this->data["ik_paysystem_alias"].':'.
            $this->data["ik_baggage_fields"].':'.
            $this->data["ik_payment_state"].':'.
            $this->data["ik_trans_id"].':'.
            $this->data["ik_currency_exch"].':'.
            $this->data["ik_fees_payer"].':'.
            $secret_key;
        $sign = mb_strtoupper(md5($sign));
        if ($this->data["ik_sign_hash"] === $sign) {
            if ($this->order->getPrice($this->data["ik_payment_id"]) == $this->data["ik_payment_amount"])
                $this->order->setDatePay($this->data["ik_payment_id"], $this->format->ts());
        }
    }
}
private function getProducts() {
    $ids = explode(",", $_SESSION["cart"]);
    $products = $this->product->getAllOnIDs($ids);
    $result = array();
    for ($i = 0; $i < count($products); $i++) {
        $result[$products[$i]["id"]] = $products[$i]["title"];
    }
    $products = array();
    for ($i = 0; $i < count($ids); $i++) {
        $products[$ids[$i]][0]++;
    }
}

```

```

        $products[$ids[$i]][1] = $result[$ids[$i]];
    }
    $str = "";
    foreach ($products as $value) {
        $str .= $value[1]. " (".$value[0]. " шт.) | ";
    }
    $str = substr($str, 0, -3);
    return $str;
}
private function getPrice() {
    $ids = explode(",", $_SESSION["cart"]);
    $summa = $this->product->getPriceOnIDs($ids);
    $value = $this->discount->getValueOnCode($_SESSION["discount"]);
    if ($value) $summa *= (1 - $value);
    return $summa;
}
}
?>

```

#### Листинг главной страницы main.tpl

```

<head>
    <title><?=$this->title?></title>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta name="description" content="<?=$this->meta_desc?>" />
    <meta name="keywords" content="<?=$this->meta_key?>" />
    <link type="text/css" rel="stylesheet" href="styles/main.css" />
    <script type="text/javascript" src="js/functions.js"></script>
</head>
<body>
    <div id="container">
        <div id="header">
            <h1>Аккаунт администратора</h1>
        </div>
        <hr />
        <?php include "menu.tpl"; ?>
        <div id="content">
            <?php include "content_". $this->content. ".tpl"; ?>
        </div>
    </div>
</body>
</html>

```

#### Листинг файла auth\_class.php

```

<?php
require_once "config_class.php";
require_once "format_class.php";

class Auth {

    private $config;
    private $format;

    public function __construct() {
        $this->config = new Config();
    }

    public function checkAdmin($login, $password, $hash = true) {
        if (!$hash) $password = $this->format->hash($password);
        $login = mb_strtolower($login);
        $real_login = mb_strtolower($this->config->adm_login);
    }
}

```

```

        return (($login === $real_login) && ($password === $this->config->adm_password));
    }
}
?>

```

#### Листинг файла adminauthcontent\_class.php

```

<?php
require_once "adminmodules_class.php";

class AdminAuthContent extends AdminModules {

    protected $title = "Вход в аккаунт администратора";
    protected $meta_desc = "Вход в аккаунт администратора.";
    protected $meta_key = "администратор, аккаунт администратора, аккаунт администратора
вход";

    public function __construct() {
        parent::__construct(false);
    }

    protected function getContent() {
        if ($this->template->auth) $this->redirect($this->url_admin->index());
        if ($_SERVER["HTTP_REFERER"] != $this->url_admin->getThisURL()) {
            if ($_SERVER["HTTP_REFERER"] != $this->url_admin->action()) {
                $_SESSION["r"] = $_SERVER["HTTP_REFERER"];
            }
        }
        $this->template->set("login", $_SESSION["login"]);
        $this->template->set("r", $_SESSION["r"]);
        return "auth";
    }

}

?>

```

#### Листинг файла admincontent\_class.php

```

<?php
require_once "adminmodules_class.php";

class AdminContent extends AdminModules {

    protected $title = "Аккаунт администратора";
    protected $meta_desc = "Аккаунт администратора Интернет-магазина.";
    protected $meta_key = "администратор, аккаунт администратор, аккаунт администратора
интернет магазин";

    protected function getContent() {
        $start = $this->format->getTime("", true);
        $end = $this->format->getTime("", false);
        $result = $this->statistics->getDataForAdmin($start, $end);
        $this->template->set("result", $result);
        return "index";
    }

}

?>

```

#### Листинг файла `admindiscountscontent_class.php`

```
<?php
require_once "adminform_class.php";

class AdminDiscountsContent extends AdminForm {

    protected $title = "Купоны";
    protected $meta_desc = "Страница с купонами";
    protected $meta_key = "купоны, список купонов";

    protected function getFormData() {
        $form_data = array();
        $form_data["fields"] = array("code", "value");
        $form_data["func_add"] = "add_discount";
        $form_data["func_edit"] = "edit_discount";
        $form_data["title_add"] = "Добавление купона";
        $form_data["title_edit"] = "Редактирование купона";
        $form_data["get"] = $this->discount->get($this->data["id"]);
        $form_data["form_t"] = "discount_form";
        $form_data["t"] = "discounts";
        $form_data["obj"] = $this->discount;
        $form_data["table_data"] = $this->discount->getTableData($this->config-
>pagination_count, $this->page_info["offset"]);
        return $form_data;
    }
}

?>
```

#### Листинг файла `adminform_class.php`

```
<?php
require_once "adminmodules_class.php";

abstract class AdminForm extends AdminModules {

    protected function getContent() {
        $form_data = $this->getFormData();
        $form = false;
        if ($this->data["func"] == "new") {
            $form = true;
            $fields = array();
            for ($i = 0; $i < count($form_data["fields"]); $i++) {
                $fields[$form_data["fields"][$i]] =
$_SESSION[$form_data["fields"][$i]];
            }
            $this->template->set("func", $form_data["func_add"]);
            $this->template->set("form_title", $form_data["title_add"]);
        }
        elseif ($this->data["func"] == "edit") {
            $form = true;
            $data_info = $form_data["get"];
            if (!$data_info) $this->notFound();
            $fields = array();
            for ($i = 0; $i < count($form_data["fields"]); $i++) {
                $fields[$form_data["fields"][$i]] =
$data_info[$form_data["fields"][$i]];
            }
            $this->template->set("func", $form_data["func_edit"]);
            $this->template->set("id", $this->data["id"]);
            $this->template->set("form_title", $form_data["title_edit"]);
        }
    }
}
```

```

    }
    if ($form) {
        foreach ($fields as $name => $value) {
            $this->template->set($name, $value);
        }
        return $form_data["form_t"];
    }
    else {
        $table_data = $form_data["table_data"];
        $this->page_info["count"] = ceil($form_data["obj"]->getCount() / $this->config-
>pagination_count);

        if ($this->page_info["count"] < $this->page_info["page"]) $this->notFound();
        $this->template->set("link_new", $this->url_admin->newElement());
        $this->template->set("table_data", $table_data);
        return $form_data["t"];
    }
}

abstract protected function getFormData();
}

```

#### Листинг файла adminmodules\_class.php

```

<?php
require_once "lib/abstractmodules_class.php";
require_once "lib/urladmin_class.php";
require_once "lib/auth_class.php";
require_once "lib/statistics_class.php";

abstract class AdminModules extends AbstractModules {

    protected $url_admin;
    protected $page_info;
    protected $statistics;

    public function __construct($check_auth = true) {
        parent::__construct();

        $this->url_admin = new URLAdmin();
        $this->statistics = new Statistics();
        $auth = $this->checkAuth();
        if ($check_auth && !$auth) $this->redirectAuth();
        $this->setMenu();
        $this->setPageInfo();
        $this->template->set("auth", $auth);
        $this->template->set("content", $this->getContent());
        $this->template->set("message", $this->message());
        $this->template->set("action", $this->url_admin->action());
        $this->template->set("title", $this->title);
        $this->template->set("meta_desc", $this->meta_desc);
        $this->template->set("meta_key", $this->meta_key);
        $this->template->set("pages", $this->getPages());
        $this->template->display("main");
    }

    private function getPages() {
        $pages = array();
        for ($i = 1; $i <= $this->page_info["count"]; $i++) {
            $pages[] = $this->url_admin->page($i);
        }
        return $pages;
    }
}

```

```

private function setPageInfo() {
    $this->page_info["page"] = isset($this->data["page"])? $this->data["page"]: 1;
    if ($this->page_info["page"] <= 0) $this->notFound();
    $this->page_info["offset"] = ($this->page_info["page"] - 1) * $this->config-
>pagination_count;
}

private function setMenu() {
    $this->template->set("index", $this->url_admin->index());
    $this->template->set("link_products", $this->url_admin->products());
    $this->template->set("link_orders", $this->url_admin->orders());
    $this->template->set("link_sections", $this->url_admin->sections());
    $this->template->set("link_discounts", $this->url_admin->discounts());
    $this->template->set("link_statistics", $this->url_admin->statistics());
    $this->template->set("logout", $this->url_admin->logout());
}

private function checkAuth() {
    $auth = new Auth();
    return $auth->checkAdmin($_SESSION["login"], $_SESSION["password"]);
}

private function redirectAuth() {
    $this->redirect($this->url_admin->auth());
}

protected function getDirTpl() {
    return $this->config->dir_tmpl_admin;
}
}

?>

```

#### Листинг файла adminorderscontent\_class.php

```

<?php
require_once "adminform_class.php";

class AdminOrdersContent extends AdminForm {

    protected $title = "Заказы";
    protected $meta_desc = "Страница с заказами";
    protected $meta_key = "заказы, список заказов";

    protected function getFormData() {
        $form_data = array();
        $form_data["fields"] = array("delivery", "product_ids", "price", "name", "phone",
"email", "address", "notice", "date_send", "date_pay");
        $form_data["func_add"] = "add_order";
        $form_data["func_edit"] = "edit_order";
        $form_data["title_add"] = "Добавление заказа";
        $form_data["title_edit"] = "Редактирование заказа";
        $form_data["get"] = $this->order->get($this->data["id"]);
        $form_data["form_t"] = "order_form";
        $form_data["t"] = "orders";
        $form_data["obj"] = $this->order;
        $table_data = $this->order->getTableData($this->config->pagination_count, $this-
>page_info["offset"]);
        for ($i = 0; $i < count($table_data); $i++) {
            $ids = explode(",", $table_data[$i]["product_ids"]);
            $products = $this->product->getAllOnIDs($ids);

```



```

        $table_data[$i]["products"] = array();
        $result = array();
        for ($j = 0; $j < count($products); $j++) {
            $result[$products[$j]["id"]] = $products[$j];
        }
        $ids_unique = array_unique($ids);
        $j = 0;
        foreach ($ids_unique as $v) {
            $table_data[$i]["products"][$j]["title"] = $result[$v]["title"];
            $table_data[$i]["products"][$j]["count"] = $this->getCountInArray($v,
$ids);
            $j++;
        }
    }
    $form_data["table_data"] = $table_data;

    if ($this->data["func"] == "new") {
        $product_ids = $_SESSION["product_ids"];
    }
    elseif ($this->data["func"] == "edit") {
        $product_ids = $this->order->getProductIDs($this->data["id"]);
    }
    $product_ids = explode(",", $product_ids);
    $products = $this->product->getAllOnIDs($product_ids);
    $result = array();
    for ($j = 0; $j < count($products); $j++) {
        $result[$products[$j]["id"]] = $products[$j];
    }
    $products = array();
    $ids_unique = array_unique($product_ids);
    $j = 0;
    foreach ($ids_unique as $v) {
        $products[$j]["id"] = $result[$v]["id"];
        $products[$j]["price"] = $result[$v]["price"];
        $products[$j]["title"] = $result[$v]["title"];
        $products[$j]["count"] = $this->getCountInArray($v, $product_ids);
        $j++;
    }
    $this->template->set("products", $products);
    $this->template->set("products_all", $this->product->getAllTable());

    return $form_data;
}

?>

```

#### Листинг файла adminproductscontent\_class.php

```

<?php
require_once "adminform_class.php";

class AdminProductsContent extends AdminForm {

    protected $title = "Товары";
    protected $meta_desc = "Страница с товарами";
    protected $meta_key = "товары, список товаров";

    protected function getFormData() {
        $form_data = array();
        $this->template->set("sections", $this->section->getAllData());
    }
}

```

```

        $form_data["fields"] = array("section_id", "pr_title", "price", "year", "country",
"director", "play", "cast", "description");
        $form_data["func_add"] = "add_product";
        $form_data["func_edit"] = "edit_product";
        $form_data["title_add"] = "Добавление товара";
        $form_data["title_edit"] = "Редактирование товара";
        $form_data["get"] = $this->product->get($this->data["id"], $this->section-
>getTableName());
        $form_data["get"]["pr_title"] = $form_data["get"]["title"];
        $form_data["form_t"] = "product_form";
        $form_data["t"] = "products";
        $form_data["obj"] = $this->product;
        $form_data["table_data"] = $this->product->getTableData($this->section-
>getTableName(), $this->config->pagination_count, $this->page_info["offset"]);
        return $form_data;
    }
}

?>

```

#### Листинг файла adminstatisticscontent\_class.php

```

<?php
require_once "adminmodules_class.php";

class AdminStatisticsContent extends AdminModules {

    protected $title = "Статистика";
    protected $meta_desc = "Аккаунт администратора Интернет-магазина.";
    protected $meta_key = "администратор, аккаунт администратор, аккаунт администратора
интернет магазин";

    protected function getContent() {
        $start = $this->format->getTime($_GET["start"], true);
        $end = $this->format->getTime($_GET["end"], false);
        $result = $this->statistics->getDataForAdmin($start, $end);
        $this->template->set("result", $result);
        $this->template->set("start", $this->format->time($start));
        $this->template->set("end", $this->format->time($end));
        return "statistics";
    }

}

?>

```

#### Листинг файла update.php

```

<?php

include_once 'lib/config_class.php';
include_once 'lib/database_class.php';

$conf = new Config;
$data = new DataBase;
$sids = $data->select('SELECT `id` FROM '.$conf->db_prefix.'products');
$file = fopen('base_export.csv', 'r');
$time = time();

while($prod = fgetcsv($file)) {
    if(in_array($prod['id'], $sids)) {

```

```

        $data->query("UPDATE `{$conf->db_prefix}products` SET `section_id` =
'{$prod['section_id']}', "
        . "`title` = '{$prod['title']}', "
        . "`price` = '{$prod['price']}', "
        . "`description` = '{$prod['description']}', "
        . "`date` = '{$time}' WHERE `id` = {$prod['id']}");
    } else {
        $data->query("INSERT INTO `{$conf->db_prefix}products` (`section_id`, `title`, `price`,
`description`, `date`)
VALUES({$prod['section_id']},{$prod['title']},{$prod['price']},{$prod['description']},{$time})");
    }
}

```

#### Листинг файла repost.php

```

<?php

if (isset($_POST['submit'])) {
    $out = "";
    $post = $_POST['link'];
    define('TOKEN',
'4853d592ee88552d9d8876d93c300e28b98d96a2c1dec7b884618cbce13df8103ec1f27beb248cf6519c0');
    $ids = [];

    if (!preg_match('<https://vk.com/.*/wall([0-9-]+)_([0-9]+)>', $post, $ids)) {
        echo '<div style="text-align: center;">Вы ввели неверный адрес<br><br>';
        echo '<a href="repost.php">Попробовать снова</a></div>';
        ;
        exit();
    }

    $offset = 0;
    $params = [
        'type' => 'post',
        'owner_id' => $ids[1],
        'item_id' => $ids[2],
        'filter' => 'copies',
        'count' => 1000,
    ];

    $req1 = 'https://api.vk.com/method/likes.getList?';
    do {
        $params['offset'] = $offset;

        $req = $req1 . http_build_query($params);

        $res = file_get_contents($req);
        $res = json_decode($res, true);

        foreach ($res['response']['users'] as $a) {
            if ($a > 0) {
                $out .= 'https://vk.com/id' . $a . PHP_EOL;
            } else {
                $out .= 'https://vk.com/club' . abs($a) . PHP_EOL;
            }
        }

        $offset += 1000;
    } while (count($res['response']['users']) > 0);

    header('Content-Type: text/plain; charset=utf8');
    header('Content-Disposition: attachment; filename=itog.txt');
    header("Content-Transfer-Encoding: binary ");
}

```

```

        echo $out;

    } else {
        ?>
        <div style="text-align: center; padding-top: 20px;">
            <form action="repost.php" method="post">
                <span style="font-size: 120%; padding-top: 20px;">Вставьте ссылку на пост:</span><br><br>
                <input style="width: 400px;" type="text" name="link"><br><br>
                <input style="background-color: #3c5; border: none; padding: 10px; color: white; font-size: 110%;
border-radius: 3px;" type="submit" name="submit" value="Скачать отчет">
            </form>
        </div>
    <?php } ?>

```

#### Листинг файла `mailer.php`

```

<?php
if (isset($_POST['send'])) {
    $list = explode(htmlspecialchars($_POST['addresses'])); //Создаём массив адресов получателей
    foreach ($list as $l) {
        $mail = new PHPMailer;
        $mail->CharSet = 'utf-8'; // Установка кодировки сообщения
        $mail->isSMTP(); // Установка авторизации SMTP
        $mail->Host = 'SB-HQ-CAS-01.sibir.fsk-ees.local'; // Установка сервера исходящей почты
        $mail->SMTPAuth = true; // Enable SMTP authentication
        $mail->SMTPAutoTLS = false; // Включать ли шифрование TLS по-умолчанию
        $mail->AuthType = 'NTLM'; // Тип аутентификации
        $mail->Username = 'kiseleva-ma'; // SMTP имя пользователя (указываем адрес почты
        до @)
        $mail->Realm = 'FSKSIBIR'; // Домен пользователя. FSKSIBIR
        $mail->Password = 'anysecretword'; // SMTP пароль
        $mail->Port = 587; // TCP port to connect to
        $mail->setFrom('kiseleva-oa@sibir.fsk-ees.ru'); // От кого
        $mail->addReplyTo('kiseleva-oa@sibir.fsk-ees.ru'); // Совпадает с для кого. Нужно для
        увеличения рейтинга письма.
        $mail->isHTML(true); // Формат исходящего сообщения
        $mail->Subject = htmlspecialchars($_POST['subj']);
        $mail->Body = htmlspecialchars($_POST['letter']);
        $mes->addAddress($l);
        $mail->send();
    }
}
?>
<!DOCTYPE html>
<html>
<head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
    <title>Авторассылка</title>
    <style type="text/css">
        body {
            margin: 50px;
        }

        table {
            border-collapse: collapse;
        }

        td {
            border: 1px solid #000;
            padding: 10px;
            vertical-align: top;
        }
    </style>

```

```

        input[type="text"] {
            width: 100%;
        }

        label.pass {
            color: #00f;
        }
    </style>
</head>
<body>
    <form name="send" action="mailer.php" method="post">
        <table>
            <tr>
                <td>
                    <h3>Почтовая рассылка</h3>
                </td>
            </tr>
            <tr>
                <td>
                    <label class="pass">Вставьте список почтовых адресов через запятую: </label><br>
                    <textarea rows="10" cols="100" name="addresses"></textarea>
                </td>
            </tr>
            <tr>
                <td>
                    <label class="pass">Введите тему сообщения: </label>
                    <input type="text" name="subj">
                </td>
            </tr>
            <tr>
                <td>
                    <label class="pass">Введите текст сообщения: </label><br>
                    <textarea rows="20" cols="100" name="letter"></textarea><br>
                    <input type="submit" name="send" value="Отправить">
                </td>
            </tr>
        </table>
    </form>
</body>
</html>

```